Mining Valuable Collaborations from Event Data Using the Recency-Frequency-Monetary Principle

Peer-reviewed author version

JOOKEN, Leen; JANS, Mieke & DEPAIRE, Benoit (2022) Mining Valuable Collaborations from Event Data Using the Recency-Frequency-Monetary Principle. In: Xavier Franch, Geert Poels, Frederik Gailly, Monique Snoeck (Ed.). Advanced information systems engineering (CAISE 2022), SPRINGER INTERNATIONAL PUBLISHING AG, p. 339 -354.

DOI: 10.1007/978-3-031-07472-1_20
Handle: http://hdl.handle.net/1942/38883

Mining valuable collaborations from event data using the Recency-Frequency-Monetary principle *

Leen Jooken[0000−0003−0836−8128], Mieke Jans[0000−0002−9171−2403], and Benoît Depaire[0000−0003−4735−0609]

Hasselt University, Martelarenlaan 42, 3500 Hasselt, Belgium
leen.jooken@uhasselt.be

Abstract. Collaborative work leads to better organizational performance. However, a team leader's view on collaboration does not always match reality. Due to the increased adoption of (online) collaboration systems in the wake of the COVID pandemic, more digital traces on collaboration are available for a wide variety of use cases. These traces allow for the discovery of accurate and objective insights into a team's inner workings. Existing social network discovery algorithms however, are often not tailored to discover collaborations. These techniques often have a different view on collaboration by mostly focusing on handover of work, resource profile similarity, or establishing relationships between resources when they work on the same case or activities without any restrictions. Furthermore, only the frequency of appearance of patterns is typically used as a measure of interestingness, which limits the kind of insights one can discover. Therefore we propose an algorithm to discover collaborations from event data using a more realistic approach than basing collaboration on the sequence of resources that carry out activities for the same case. Furthermore, a new research path is explored by adopting the Recency-Frequency-Monetary (RFM) concept, which is used in the marketing research field to assess customer value, in this context to value both the resource and the collaboration on these three dimensions. Our approach and the benefits of adopting RFM to gain insights are empirically demonstrated on a use case of collaboratively developing a curriculum.

Keywords: Collaboration network · mining resource behavior · RFM · social network analysis.

1 Introduction

Collaborative work leads to better organizational performance in terms of efficiency and quality when the teams are well implemented, managed and supported [25, 11]. This requires that team leaders have accurate insights into the collaboration characteristics, in order to improve team effectiveness through

* Supported by the Special Research Fund (BOF19OWB10) of Hasselt University.

team-oriented interventions [11]. However, a team leader's view on how collaboration is taking place does not always match reality [18, 23]. Research has shown that people have difficulties in accurately perceiving the informal structure of groups and that there is a negative relationship between the hierarchy level and the accuracy of perception [18, 10, 7].

Due to the increasing digitization however, more digital traces on collaboration have become available as teamwork is increasingly supported by communication and coordination tools [9]. The global COVID pandemic has further sped up the adoption of these information systems that support online collaboration out of necessity to keep businesses running. This shift provides a great opportunity as the data in these systems provide a more objective and complete view on the work that actually took place in reality.

In this study we aim to extract the resources (the actors in the team) and the collaboration relationships between these resources (based on the objects they worked on) from this type of data. This work is related to existing work in the field of Organisational Network Analysis (ONA) [9, 28], the organisational perspective in Process Mining [1], and Developer Social Networks (DSN) [12], but these are all subject to some limitations when applied to this collaboration context. The main shortcomings are that the data from collaboration systems is not suitable to use with these existing techniques, and that the existing techniques have a view on collaboration that differs from the view adopted in this paper. This will be further elaborated on in Section 2. Furthermore, for all these research areas there is often no clarity or agreement on what constitutes a valuable resource or a valuable relationship. Therefore we will expand on related work by also assessing the value of a resource and a relationship in the collaborative context, based on the Recency-Frequency-Monetary (RFM) model [6] presented in marketing literature. This model is a widely used tool designed to measure customer value, which, given an event of interest, measures how recent the event occurred (R), how frequently it occurred (F) and the monetary aspect of the event (M) [22]. Since the concept of an event can be interpreted broadly, the RFM model can be adapted to many different contexts [22]. This is usually done by modifying one or more RFM segmentation variables, such as redefining, adding or excluding variables [15]. This concept has already been successfully applied in other data mining applications to find interesting patterns, not solely based on their frequency of appearance (see Section 2).

Therefore the contribution of this paper is twofold: (1) we provide an algorithm that, using realistic constraints, can uncover how collaboration takes place in reality. The input data can come from any information system that captures digital traces of collaboration, as long as the resources, the objects of collaboration, and the timestamps of when a resource worked on an object can be extracted from it. The output is the set of resources and the set of collaboration relationships between these resources, which could be represented as a network. Furthermore (2) we substantiate the value of resources and their relationships by adopting and redefining the RFM model to gain valuable insights.

The remainder of this paper is structured as follows. In Section 2 the relevant related work on ONA, Process Mining, DSN and RFM-enriched networks is discussed. Section 3 elaborates on the algorithm's design and implementation, and formalization of the RFM concepts in the context of collaboration. A demonstration on a use case is given in Section 4. Finally, this paper is concluded in Section 5.

2 Related Work

2.1 Organizational Network Analysis

Organizational Network Analysis (ONA) is a type of Social Network Analysis (SNA) that tries to uncover and provide insights into relationships between people in an organisation [9, 28]. There exists a broad array of research within this domain, as it covers all sorts of different networks. Research on collaborative networks has covered actors in movies [17], co-authors on research papers [17], software developers on a project [17, 12, 4], problem-solving collaborations [9] and more. A downside of the ONA approach is that it requires the input data to explicitly state the collaboration relationships (usually collected through surveys [9]), which is not necessarily the case for the data in these collaboration systems. Often only information on who worked on what and when this happened, is recorded in this data. Therefore firstly, a method is needed that can discover collaboration relationships from this type of data. The two other domains on the other hand, can handle this type of input data, but have shortcomings in other areas. As for the value of resources and their relationships, ONA provides metrics such as degree, closeness and betweenness centrality, and modularity [28]. One could for example derive the core members and boundary spanners, but these insights are limited to the network structure, not the actual importance of a resource or a relationship for collaboration.

2.2 Process Mining

The goal of the field of Process Mining is to turn data on events in a process, collected by process-aware information systems, into insights and actions [1]. The lion's share of research in this field is focused on control-flow discovery [26, 16], which aims to discover a model that best represents the process in terms of activities and their dependencies [1]. There has been some attention, albeit limited, to the organizational perspective of Process Mining, where the focus lies on the resources that carry out the activities in the process [1]. The view on collaboration that is put forward in this literature is strongly intertwined with the process context and therefore differs from the view on collaboration adopted in this paper. In this paper, a collaboration relationship between resources is assumed when they work on the same objects in close proximity in time. In the Process Mining field a relationship between resources is not established on object level, but either on the level of working on the same case (i.e. the specific

process instance) or activities [2, 26, 3, 23, 24]. Therefore analyses mostly focus on handover of work networks [2, 26, 3] or resource profile similarity [3]. As for the value of relationships in these networks, if this value exists at all, it is based on its frequency of appearance [2, 3]. The resources itself are almost never given an importance value, with the exception of the work of Pika et al. [23] where resource utilization and productivity is analysed.

2.3 Developer Social Networks

Lastly there is a large body of work dedicated to Developer Social Networks (DSN), which focuses on constructing social networks that represent the collaborative effort of writing software code [12]. Often however, the most commonly used definition of collaboration is very lax, stating that two developers are connected when they worked on the same code file, without taking into account any constraints [4, 5, 29]. In reality however, collaboration is often more nuanced than this. Therefore some efforts have been made to refine this definition (some of them we will also adopt in this work): by using a time window in which collaboration must take place [19]; taking a granularity measurement into account so that collaboration on one object differs from that on a dozen objects [19]; incorporating an object importance value [13]; or adopting similarity measures to establish relationships between two resources [13]. Another well known problem is that developers can choose how often they log their work: if they log it in 1 big chunk or several small chunks. Therefore the number of times a developer worked on a code file is in itself not an objective metric and cannot be used in calculations or comparisons without some kind of modification. This will also be taken into account in our work by grouping the work of a resource into work sessions.

As for the value of resources and relationships: the resources in DSN literature only very seldomly get assigned a weight value, whereas the collaboration relationships get assigned a weight value in some studies [13, 19, 4, 27], but the methods vary and there is no universal agreement. The value of a relationship is sometimes given as part of the survey data [27], or calculated using a similarity or distance metric [13]. However we criticize that similarity between developers does not necessarily entail collaboration.

2.4 Recency-Frequency-Monetary model

The RFM model [6] is a widely used tool designed to measure customer value, and therefore mostly used for customer segmentation and to predict customer churn, retention, loyalty and profitability [8]. The RFM model has been combined with data mining techniques, such as sequential pattern mining [8], bayesian networks [20] and deep neural networks [14] for prediction and pattern mining tasks. To the best of our knowledge there are no studies that incorporate the RFM dimensions in a social network discovery algorithm. There are however several studies that enrich pre-mined network representations with RFM information [30, 20–22, 14]. The work of Mitrović et al. [20–22] lies closest to the

work presented in this paper. It focuses on enriching telecom call graphs with RFM information in order to include both interaction and structural features as explanatory variables for churn prediction. Their work covers two approaches: an RFM-embedded and an RFM-augmented graph. In the first approach the relationship between two customers is given a weight based on the distance between the RFM vectors of both customers. This differs from the approach in this paper as we calculate the RFM values for the edge directly and do not combine these into a summarized score, hence losing information. In their second approach Mitrović et al. augment the network by adding the RFM information as nodes to the network itself, hence changing its topology. Here no RFM value is calculated for the relations between the customers. The approach in this paper refrains from changing the topology of the network and provides RFM values for the relationships as well.

To the best of our knowledge, no studies adopting the RFM model for a network representation in the context of collaboration between resources have been carried out yet.

3 Algorithm Design

This section elaborates on the input data requirements and the different steps taken to discover collaboration relationships between resources and to value both the relationships and the resources on the three RFM dimensions.

First of all, it is important to note that the assumption is made that a resource can log their work as frequently as desired, as based on and justified in Section 2.3. This means that a resource can choose to register their work often in small chunks or less often in big chunks, and therefore the amount of times work is registered does not necessarily entail how often a resource has worked or collaborated. To tackle this we introduce the concepts of a work session and a collaboration session, which group work that is registered in close proximity in time together. The high-level algorithm design is then as follows. First the collaboration relationships between resources are mined from the data, making use of the concept of a collaboration session. Next, the value of such a relationship on the RFM dimensions is calculated. To also calculate these values for a resource, the preparatory step in which the resource's work is divided into work sessions is carried out first. The RFM model is redefined for this collaboration context as follows. The recency value is based on how recent a resource worked or how recent a collaboration took place. The frequency value indicates how frequent a resource worked or collaborated, and the monetary value indicates the importance of a resource or collaboration based on the importance value of the objects that were worked on. These different steps will all be elaborated on in the next subsections.

3.1 Input requirements

The algorithm's input data can come from any information system that captures collaboration relationships, as long as it is possible to extract a set of events

in which a resource worked on an object at a specific moment of time. The accepted input data structure is therefore an event log, a concept adopted from the Process Mining field [1]. A typical Process Mining event log is constructed as a list of events, in which each event must specify a case ID (i.e. a specific instance of a process) and an event ID. Furthermore information on how to (partially) order events in time must be present. Additional attributes such as an activity label, the exact timestamp and the resource that carries out the activity are optional, but appear in most event logs in practise. The input requirement for our algorithm differs in the way that the case ID is not required, but the exact timestamp, the resource and the object of the activity are. This is formalized in Definition 1.

Definition 1. *An event is defined as a tuple (event ID, resource A, object O, timestamp T) and represents a specific point in time T when resource A worked on object O.*

3.2 Mining the collaboration relationships

The existence of a collaboration relationship between two resources depends on whether these resources engaged in a collaboration session, as stated in Definition 2 and 3. Note that all objects that both resource A and B worked on are considered when determining if a collaboration relationship between these two resources exists. To determine the set of collaboration sessions between resource A and B on an object O (Algorithm 1), two user-specified parameters are required:

- A minimal time value t_{min} in minutes, indicating that if the time between two consecutive events exceeds this threshold, these events certainly belong to separate collaboration sessions
- The maximum length of a collaboration session t_{max} in minutes

The set of collaboration sessions between two resources is further also used in Section 3.3 to calculate the RFM values for a collaboration relationship.

Definition 2. *There exists a collaboration relationship between two resources A and B if and only if there exists $\geq$ 1 collaboration session between these two resources.*

Definition 3. *A collaboration session between resources A and B is a time window with size $\leq t_{max}$ that contains $\geq$ 1 event in which A worked on an object O, and $\geq$ 1 event in which B worked on that same object O. The time between 2 consecutive events in this window is always $\leq t_{min}$. The set of collaboration sessions between resources A and B on an object O is calculated as stated in Algorithm 1.*

Algorithm 1: Get collaboration sessions for resource pair (A,B) on object O

Input : The set of events in which resource A worked on object O ∪ the set of events in which resource B worked on object O, t_{min}, t_{max}
Output : Set of collaboration sessions for resource pair (A,B) on object O

```
1  timestamps ← order timestamps of all the events chronologically;
2  t_current ← timestamps [0];
3  e_current ← corresponding event of t_current;
4  session [e_current ];
5  for i ← 1 to len(timestamps) do
6  |   t_next ← timestamps [i];
7  |   e_next ← corresponding event of t_next;
8  |   if t_next − t_current > t_min then
9  |   |   ProcessCollabSession(session,t_max) ;          /* See Algorithm 2 */
10 |   |   session ← [ ]
11 |   end
12 |   append e_next to session;
13 |   t_current ← t_next;
14 |   e_current ← e_next;
15 end
16 ProcessCollabSession(session,t_max) ;                  /* See Algorithm 2 */
```

Algorithm 2: ProcessCollabSession($session, t_{max}$)

Input : A set of events with as resource A OR B and as object O, t_{max}
Output : Set of collaboration sessions for resource A and B on object O

```
1  timestamps ← order timestamps within this session chronologically;
2  t_first ← timestamps [0];
3  t_last ← timestamps [-1];
4  if t_last − t_first > t_max then
   |   /* Session needs to be split up in parts                                */
5  |   window1, window2 ← find the largest gap between 2 consecutive timestamps in this
   |     window and split here the window into 2 parts;
   |   /* if there are multiple options choose the one that lies closest to the midpoint
   |      of the current window under consideration                            */
   |   /* process these 2 new session windows by recursively calling this function again
   |      */
6  |   ProcessCollabSession(window1, t_max);
7  |   ProcessCollabSession(window2, t_max);
8  else
   |   /* This session is not too long, check if collaboration took place      */
9  |   if session contains ≥ 1 event which has A as the resource AND ≥ 1 event which has
   |     B as the resource then
10 |   |   collaborationsessions ← save this window with the including events as one
   |   |     collaboration session to a global container;
11 |   end
12 end
```

3.3 RFM values for a relationship

To calculate the values of the three RFM dimensions for a collaboration relationship between resources A and B, their set of collaboration sessions is required. This set is obtained by taking the union of the sets of collaboration sessions on every object they collaboratively worked on, as discussed in Section 3.2.

Recency

Definition 4. *The recency value of a collaboration relationship between resource A and B is an indication of how recent their collaboration sessions fall on the timeline of the log that is under consideration, calculated as indicated in Algorithm 3.*

To calculate the recency value, the timeline of the event log that is examined is divided into time windows of a predefined width (user-specified parameter *'windowsize'*). These windows are numbered starting from the least recent one (number 1) to the most recent one (number n, with n the total number of windows). Next, each window gets assigned a weight equal to their window number over the total number of windows. This results in the least recent one having a weight of 1/n, to the most recent one having a weight of 1. The collaboration sessions of a resource pair then get assigned to their corresponding windows, based on the median timestamp of a session's included events. The recency value of this collaboration relationship is then the relative number of items (number / total number of collaboration sessions) in a window times the window weight, and this summed over all the windows. This is formalized in Algorithm 3.

Algorithm 3: Recency(resource A, resource B)

Input : Set of all collaboration sessions between resource A and B, windowsize as the size of the bins, t_{first} the first timestamp of the entire project event log, t_{last} the last timestamp of the entire project event log
Output : Recency value for the collaboration relationship between resource A and B

```
   /* Divide the timeline into bins                                          */
 1 bins ← divide the timeline between t_first and t_last into bins of width windowsize starting
     from t_last and working towards t_first;
   /* Give each bin a weight                                                 */
 2 for i ← 1 to len(bins) do
 3 |   binweights [i] ← i/ len(bins);
 4 end
   /* Get for each collaboration session the median time value for the included events;
      these become the data points to bin                                    */
 5 datapoints ← [ ];
 6 for i ← 1 to len(collaborationsessions) do
 7 |   timestamps ← list of timestamps of all the events in collaborationsessions[i];
 8 |   datapoints [i] ← median(timestamps);
 9 end
10 add each data point from datapoints into the appropriate bin from bins;
   /* Calculate the relative frequency of data points in each bin            */
11 for i ← 1 to len(bins) do
12 |   bincount ← # datapoints in bins [i];
13 |   relativeBinCount [i] ← bincount / len(datapoints);
14 end
   /* Calculate final recency value                                          */
15 recency ← 0;
16 for i ← 1 to len(relativeBinCount) do
17 |   recency + ← relativeBinCount [i] · binweights [i];
18 end
19 return recency
```

Frequency

Definition 5. *The frequency value of a collaboration relationship between resource A and B is defined as the total number of collaboration sessions that exist between A and B, taking into account all the objects they worked on, as calculated using Algorithm 1 in Section 3.2.*

$$F(A,B) = \sum_{O} \#\ collaboration\ sessions(A,B,O) \tag{1}$$

Monetary The monetary value of a collaboration relationship represents the importance of the work package, i.e. the collection of objects, both resources collaboratively worked on. In order to calculate this value a notion of *"**the importance value of an object**"* is necessary. These object importance values are highly project-dependent and therefore it is difficult to provide a default method of calculation that makes sense in all cases. Therefore these values can best be included in the event log as attributes, or calculated based on a user-defined function.

The method that was chosen for the demonstration in Section 4 is adopted from the software development context. An object (or file in that context) is considered important for collaboration if it continues to *"grow over time"*. Files that get altered regularly are good candidates for collaboration since multiple people having knowledge of them secures their further evolution. Based on these assumptions, Formula 2 is used to calculate the object importance. To ensure that objects that have been around for a long time are not favored over relatively new ones, we work with a ratio that takes the life span of the object into account. (Note that in this case information on the creation and deletion of an object must be available.) This also results in a larger importance value for objects created towards the end of the event log, however they are considered important as they are most relevant at this very moment in the project stage.

$$I(O) = \frac{\#\ months\ in\ which\ O\ got\ worked\ on}{\#\ months\ O\ existed} \tag{2}$$

Definition 6. *The monetary value of a collaboration relationship between resource A and B is defined as the number of collaboration sessions on an object O times the importance value of that object, and this summed over all the objects the pair of resources (A,B) collaborated on.*

$$M(A,B) = \sum_{O} \#\ collaboration\ sessions(A,B,O) \cdot Importance(O) \tag{3}$$

The number of collaborations on an object is included in the equation to distinguish between a pair that worked once on an important object and often on less important objects, compared to a pair that constantly worked on an important object and seldom on less important ones. If we would not take the number of collaborations into account, both pairs would have the same monetary value, while the latter should actually have a bigger monetary value.

3.4 Constructing the work sessions

As highlighted in the beginning of Section 3, in order to calculate the RFM values for a resource, the resource's work package must first be divided into work sessions. Such a work session consists of several events in which the resource worked on any object, that are grouped together when they occur in close proximity in time.

Definition 7. *A work session of resource A is a time window with size* $\leq t_{max}$ *that contains* ≥ 1 *event, with all these events having A as the resource, and the time between 2 consecutive events in this window is always* $\leq t_{min}$*.*

The set of work sessions of a resource A is calculated almost identical to Algorithm 1, with two differences. First, the set of events as input just consists of all the events that have A as the resource (regardless of the object that was worked on). Note that this means that the work sessions are not calculated per object, as was the case for the collaboration sessions. Secondly, the check if collaboration took place becomes redundant, which means that the session always gets saved as a work session.

3.5 RFM values for a resource

The work sessions that were calculated for a resource in the previous section will serve as the starting point for the calculation of the resource's RFM values, similar to the approach for the relationships.

Recency

Definition 8. *The recency value of a resource A is an indication of how recent their work sessions fall on the timeline of the log that is under consideration. This value is calculated in the same way as the recency value for a collaboration relationship, as indicated in Algorithm 3, with the difference that instead of the collaboration sessions between two resources, the set of all work sessions of resource A is used as input.*

Frequency

Definition 9. *The frequency value of a resource A is defined as the total number of work sessions identified for resource A, as explained in Section 3.4.*

$$F(A) = \#\ work\ sessions(A) \tag{4}$$

Monetary The monetary value of a resource represents the importance of their work package. There are different metrics that can be used to calculate this value, depending on the end user's goal. The method presented here is analogous to the reasoning followed to calculate the monetary value of a relationship in Section 3.3. Possible alternative methods include: using the betweenness centrality

when the interest lies in cross-functional team members; or using the eigenvector centrality to emphase resources that are very central in the team.

Definition 10. *The monetary value of a resource A is defined as the number of work sessions that include ≥ 1 event that has O as object, times the importance value of that object, and this summed over all the objects resource A worked on.*

$$M(A) = \sum_{O} \#\ work\ sessions(A)\ that\ include\ O \cdot Importance(O) \quad (5)$$

4 Demonstration

The tool is implemented in Python and available on Github[1]. In this section a demonstration of the tool and possible interesting insights will be provided. As a use case the "Machine Learning for Beginners curriculum"[2] project is used, which is an initiative of the Azure Cloud advocates at Microsoft. This team of authors, illustrators and Microsoft Student Ambassadors has come together to create a freely available 26-lesson curriculum on machine learning. Furthermore it is open to the public to contribute translations, fix bugs, or suggest and provide new lessons. The incremental traces of collaboratively developing the lessons and translations are available for study. The data was extracted from GitHub, which is a version control system mainly used for software development projects. However, it also harbors collaboration information on other topics such as this use case of developing a curriculum. Do note that the tool's strength lies in its applicability to data from any information system that captures digital traces of collaboration, beyond only GitHub.

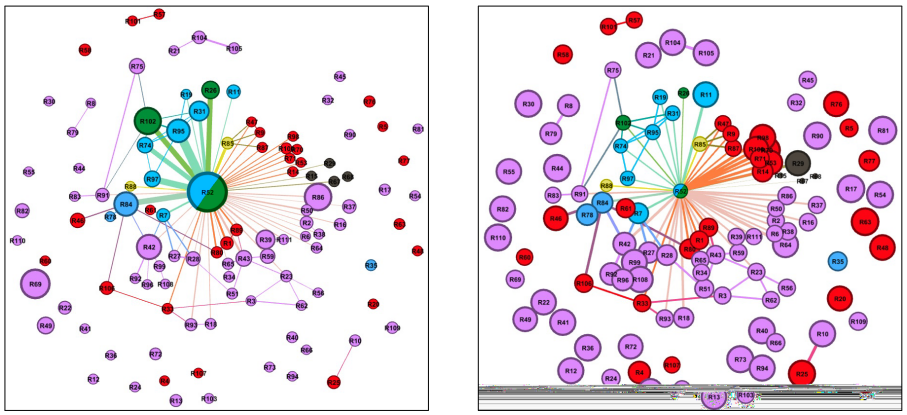
An event log consisting of events that describe the timestamp when a resource worked on a file (i.e. the object of collaboration) was extracted based on the log of Git commits. The resulting event log analyzed in this demo includes the work between January 31 2021 and November 13 2021. The parameter settings for the demonstration were set to t_{min} and t_{max} respectively equal to 2 weeks and to 7 days for the discovery of the collaboration sessions; and 24 hours and 4 hours for the identification of the work sessions. The window size for the recency calculation for both a resource and a collaboration relationship was set to 24 hours. All the resources' names are anonymized in the discussion to maintain their privacy.

The collaboration network that was discovered using our tool is shown in Figure 1. The color codes were added manually based on the available project information. To analyze the RFM values for both the resources and relations, the recency, frequency, and monetary ranges are divided into five segments, similar to how RFM has been traditionally used for market segmentation in the marketing field [8]. The resulting segments for each dimension and the number of elements

[1] https://github.com/LeenJooken/RFMCollaborationMiner
[2] https://github.com/microsoft/ML-For-Beginners

that fall within each segment are shown in Table 1. This table will be used to highlight a selected number of insights, by discussing groups that are formed using a set intersection of segments from each dimension. For example, resource group 1 - x - 4/5 refers to the all the resources that have a recency value in segment 1 ([0.00, 0.25[), any possible frequency value, and a monetary value in segment 4 or 5 ($\geq$ 20), as described below.

$$\begin{aligned} \textit{Resource group } 1-x-4/5 \ = \ & \forall \textit{ resources} \in \textit{segment } R1 \cap \\ & (\ \forall \textit{ resources} \in \textit{segment } M4 \ \cup \ \forall \textit{ resources} \in \textit{segment } M5 \ ) \end{aligned} \tag{6}$$



(a) Monetary values as weights (b) Recency values as weights

Fig. 1: The mined collaboration network with as node and edge weights respectively the monetary value (a) and the recency value (b). Therefore in (a) bigger nodes and thicker edges represent higher monetary values, in (b) they represent higher recency values. The color codes: author (blue), illustrator (green), MS student (yellow), bug fixer (red), translator (purple), bot (black).

First of all, the core resources of the network, that are mostly authors and illustrators, are characterized by the highest monetary values (segments 3 to 5) (Figure 1 (a)), mostly also high frequency values (segments 4 and 5), but very low recency values (segment 2) (Figure 1 (b)), (resulting in group 2 - 4/5 - 3/4/5). This means that they worked often and that their work package is important, but that this work took place in the beginning of the project. This is also reflected in their internal collaboration relationships that are not recent, but have a large monetary value (group 2 - x - 4/5). Figure 1 (b) shows then that the resources with recency values in the highest segment (group 5 - x - x) are all translators

Table 1: Segmentation of the RFM dimensions. The number of elements (resources or collaboration relationships) that fall within each segment for each dimension is indicated between brackets. (Sums to total number of elements by column.)

#	Resource			Collaboration relationship		
	R	**F**	**M**	**R**	**F**	**M**
1	[0.00, 0.25[(3)	[0, 2 [(60)	[0, 3 [(69)	[0.00, 0.25[(2)	[0 - 3 [(80)	[0 - 0.6 [(23)
2	[0.25, 0.60[(40)	[2, 5 [(30)	[3, 10[(28)	[0.25, 0.50[(15)	[3 - 5 [(10)	[0.6 - 2 [(64)
3	[0.60, 0.75[(24)	[5, 10[(10)	[10, 20[(7)	[0.50, 0.65[(51)	[5 - 10[(6)	[2 - 5 [(9)
4	[0.75, 0.90[(17)	[10, 25[(9)	[20, 50[(6)	[0.65, 0.80[(19)	[10 - 20[(6)	[5 - 10[(5)
5	[0.90, 1.00] (27)	$\geq$25 (2)	$\geq$50 (1)	[0.80, 1.00] (16)	$\geq$20 (1)	$\geq$10 (2)

or bug fixers. This makes sense as the lessons first had to be created before they were made available for translations and fixes. The 4th recency segment (group 4 - x - x) contains one author that added a lesson at a later stage in the project (Figure 1 (b) resource R11). Next, it is easy to notice that resource R52 stands out above all in terms of monetary and frequency value (group x - 5 - 5) and turns out to act as the lead for this project. This resource is also involved in the most collaboration relationships, namely 48 (with the second highest being 6 for reference). If we further look at resources with a high monetary value (group x - x - 4/5) (Figure 1 (a)), we notice that resources R69 and R86 barely engage in collaborations (0 and 1 respectively), which makes them crucial resources for further knowledge retention of this project. Lastly, if we take a look at group 1-1-1, it shows that this group consists of three bot resources (microsoft open source, microsoft-github-operations[bot] and azure static web apps) that were all only used by the lead resource R52 to initiate the project. These are depicted as black nodes in Figure 1.

5 Conclusions, Limitations, and Future Work

Insights into a team's collaborative characteristics are essential to improve organizational performance. Nowadays, accelerated by the rise of COVID, more and more digital traces on all sorts of collaboration are available as collaborative work is increasingly supported by information systems. In this study we explored the potential of this data to provide realistic and valuable insights into collaborative work. Existing methods from the domains of Organizational Network Analysis, Process Mining and Developer Social Networks are subject to certain shortcomings when applied to this collaboration context. Therefore in this study we presented an algorithm designed to mine collaboration relationships between resources from event data extracted from these systems, which captures the exact timestamp of when a resource worked on which object. Furthermore, we expanded on existing literature by exploring a new research path on how to value

a resource and a relationship, by adopting and redefining the RFM model from marketing research. This model allows to gain insights beyond how frequently collaboration took place, by also providing a recency and monetary dimension. The algorithm was demonstrated on a use case of collaboratively and incrementally developing a curriculum on machine learning. The demonstration showed that insights into the general structure of the collaboration network could be provided, as well as insights into how resources and relationships are positioned on the different RFM dimensions.

There are however some limitations that should be addressed. The work presented in this paper starts from the premise that a project is available for analysis that clearly indicates the objects on which collaboration took place. This will not always be the case when data is extracted from information systems, and often data cleaning and preprocessing might be required. Further, the method that was chosen to calculate the importance value of an object for the demonstration in Section 4 values objects based on their importance for collaboration and not necessarily their business value. This solution could be appropriate for several use cases. However, note that if applied to a traditional software development project, the method should be fine-tuned, as the current one could point to objects that required frequent bug fixes and not necessarily those objects that are the most valuable in terms of feature criticality.

To conclude, there are several possible extensions to this work that could be addressed in the future. First of all, a validation study on a use case with expert feedback is planned for the future. Next, it may be difficult for an end-user to provide (the optimal) parameter values. Possible future directions may be working with fuzzy constraints instead of hard boundaries when calculating the work and collaboration sessions; or providing a method that calculates an optimal default parameter setting. Furthermore, different methods to determine the importance value of an object might be explored. Lastly, the methods of calculating a work or collaboration session could be handled in an alternative way by positioning them as optimization problems to find the most optimal grouping of events in sessions. Examining the effects of repositioning these methods could be interesting further research.

References

1. van der Aalst, W.: Process mining: Data science in action (2016), ISBN 978-3-662-49851-4
2. van der Aalst, W.M.P., Reijers, H.A., Song, M.: Discovering Social Networks from Event Logs. Computer Supported Cooperative Work (CSCW) **14**(6), 549–593
3. van der Aalst, W.M.P., Song, M.: Mining Social Networks: Uncovering Interaction Patterns in Business Processes. In: Desel, J., Pernici, B., Weske, M. (eds.) Business Process Management. pp. 244–260. Lecture Notes in Computer Science, Springer
4. Aljemabi, M.A., Wang, Z.: Empirical study on the similarity and difference between vcs-dsn and bts-dsn. In: Proceedings of the 2017 International Conference on Management Engineering, Software Engineering and Service Sciences. p. 30–37. ICMSS '17, Association for Computing Machinery, New York, NY, USA (2017)

5. Bird, C., Pattison, D., D'Souza, R., Filkov, V., Devanbu, P.: Latent social structure in open source projects. In: Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of Software Engineering. p. 24–35. SIGSOFT '08/FSE-16, Association for Computing Machinery (2008)
6. Bult, J.R., Wansbeek, T.: Optimal Selection for Direct Mail. Marketing Science **14**(4), 378–394
7. Casciaro, T.: Seeing things clearly: social structure, personality, and accuracy in social network perception. Social Networks **20**(4), 331–351 (1998)
8. Chen, Y.L., Kuo, M.H., Wu, S.Y., Tang, K.: Discovering recency, frequency, and monetary (RFM) sequential patterns from customers' purchasing data. Electronic Commerce Research and Applications **8**(5), 241–251 (oct 2009)
9. Cross, R., Ehrlich, K., Dawson, R., Helferich, J.: Managing Collaboration: IMPROVING TEAM EFFECTIVENESS THROUGH A NETWORK PERSPECTIVE. California Management Review **50**(4), 74–98
10. Cullen, K.L., Palus, C.J., Appaneal, C.: Developing Network Perspective: Understanding the Basics of Social Networks and their Role in Leadership [White paper]. Tech. rep., Center for Creative Leadership (2014). https://doi.org/10.35613/ccl.2014.1019
11. Guzzo, R.A., Dickson, M.W.: TEAMS IN ORGANIZATIONS: Recent Research on Performance and Effectiveness. Annual Review of Psychology **47**(1), 307–338
12. Herbold, S., Amirfallah, A., Trautsch, F., Grabowski, J.: A systematic mapping study of developer social network research. Journal of Systems and Software **171**, 110802 (2021)
13. Jermakovics, A., Sillitti, A., Succi, G.: Mining and visualizing developer networks from version control systems. In: Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering. p. 24–31. CHASE '11, Association for Computing Machinery, New York, NY, USA (2011)
14. Lai, C.Y., Li, Y.M., Lin, L.F.: A social referral appraising mechanism for the e-marketplace. Inf. Manage. **54**(3), 269–280 (apr 2017)
15. Li, J., Cao, S.: The study on high-value user identification of localized information platform. Journal of Physics: Conference Series **1883**(1), 012109 (apr 2021)
16. Ly, L.T., Rinderle, S., Dadam, P., Reichert, M.: Mining Staff Assignment Rules from Event-Based Data. In: Bussler, C.J., Haller, A. (eds.) Business Process Management Workshops. pp. 177–190. Lecture Notes in Computer Science, Springer
17. Madey, G., Freeh, V., Tynan, R.: THE OPEN SOURCE SOFTWARE DEVELOPMENT PHENOMENON: AN ANALYSIS BASED ON SOCIAL NETWORK THEORY. In: AMCIS 2002 Proceedings. pp. 1806–1813. Association for Information Systems
18. Mehra, A., Smith, B.R., Dixon, A.L., Robertson, B.: Distributed leadership in teams: The network of leadership perceptions and team performance. The Leadership Quarterly **17**(3), 232–245
19. Meneely, A., Williams, L.: Socio-technical developer networks: Should we trust our measurements? In: Proceedings of the 33rd International Conference on Software Engineering. pp. 281–290. ICSE '11, Association for Computing Machinery
20. Mitrovic, S., Baesens, B., Lemahieu, W., De Weerdt, J.: tcc2vec: Rfm-informed representation learning on call graphs for churn prediction. Information Sciences **557**, 1–16 (2019)
21. Mitrovic, S., De Weerdt, J.: Dyn2Vec: Exploiting dynamic behaviour using difference networks-based node embeddings for classification. In: Proceedings of the International Conference on Data Science. pp. 194–200. CSREA Press

22. Mitrovic, S., Singh, G., Baesens, B., Lemahieu, W., De Weerdt, J.: Scalable rfm-enriched representation learning for churn prediction. vol. 2018-January, pp. 79–88. IEEE (2017)
23. Pika, A., Leyer, M., Wynn, M.T., Fidge, C.J., Hofstede, A.H.M.T., Aalst, W.M.P.V.D.: Mining Resource Profiles from Event Logs. ACM Transactions on Management Information Systems **8**(1), 1:1–1:30
24. Pika, A., Wynn, M.T., Fidge, C.J., ter Hofstede, A.H.M., Leyer, M., van der Aalst, W.M.P.: An Extensible Framework for Analysing Resource Behaviour Using Event Logs. In: Jarke, M., Mylopoulos, J., Quix, C., Rolland, C., Manolopoulos, Y., Mouratidis, H., Horkoff, J. (eds.) Advanced Information Systems Engineering. pp. 564–579. Lecture Notes in Computer Science, Springer International Publishing
25. Recardo, R., Wade, D., Mention, C.: Teams: Who Needs Them and Why? Teams: Who Needs Them and Why?, Gulf Publishing Company (1996)
26. Song, M., van der Aalst, W.M.P.: Towards comprehensive support for organizational mining. Decision Support Systems **46**(1), 300–317
27. Tymchuk, Y., Mocci, A., Lanza, M.: Collaboration in open-source projects: Myth or reality? In: 11th Working Conference on Mining Software Repositories, MSR 2014 - Proceedings (2014)
28. Wasserman, S., Faust, K., et al.: Social network analysis: Methods and applications. Cambridge university press (1994), ISBN 0-521-38707-8
29. Wolf, T., Schroter, A., Damian, D., Nguyen, T.: Predicting build failures using social network analysis on developer communication. In: 2009 IEEE 31st International Conference on Software Engineering. pp. 1–11 (2009)
30. Xue, Y., Chen, J., Zhou, Y.: Research on user discovery based on loyalty in sns. In: Proceedings of the 2017 International Seminar on Social Science and Humanities Research (SSHR 2017). pp. 399–406. Atlantis Press (2017/12)