Finding the "Liberos": Discover Organizational Models with Overlaps

Jing Yang[1], Chun Ouyang[2], Maolin Pan[1], Yang Yu[1](✉), and
Arthur H.M. ter Hofstede[2]

[1] Sun Yat-sen University, Guangzhou, China
yangj357@mail2.sysu.edu.cn, {panml,yuy}@mail.sysu.edu.cn
[2] Queensland University of Technology, Brisbane, Australia
{c.ouyang, a.terhofstede}@qut.edu.au

Abstract. Organizational mining aims at gaining insights for business process improvement by discovering organizational knowledge relevant to the performance of business processes. A key topic of organizational mining is the discovery of organizational models from event logs. While it is common for modern organizations to have employees sharing roles and responsibilities across different internal groups, most of the existing methods for organizational model discovery are unable to identify such overlaps. The overlapping resources are likely to be generalists in an organization. Existing findings in process redesign best practices have proven that generalists can help increase the flexibility of a business process (similarly to the flexibility of the role of "libero" in certain team sports). In this paper we propose an approach capable of discovering organizational models with overlaps and thus helping identify generalists in an organization. The approach builds on existing cluster analysis techniques to address the underlying technical challenges. Through experiments on real-life event logs the applicability and effectiveness of the proposed method are evaluated.

Keywords: Process mining · Organizational mining · Organizational model mining · Overlapping clustering

1 Introduction

Process mining enables data-driven process analysis using the massive amount of event log data captured by information systems in today's organizations. Various techniques have been developed to help extract insights about the actual business processes with the ultimate goal to improve process performance as well as the organizations' business performance. While the main focus of process mining is on the control-flow perspective, recent years have seen research devoted to mining other aspects such as the organizational context of business processes.

Organizational mining focuses on discovering organizational knowledge, including e.g. organizational structures and human resources relevant to the performance of a business process, from event log data [1]. In any organization where humans play a dominant role, organizational mining helps managers gain a better understanding of the *de facto* grouping of human resources and their interactions thus to improve the related business processes. The importance of such organizational knowledge in process improvement is also emphasized by the fact that 10 of the 29 best practices in process redesign proposed in [2] are concerned with the structure and population (i.e. resources) of an organization.

Hence, an interesting research topic concerns the discovery of organizational models from event log data. Given the fact that in many real-life event logs, only limited information about process execution is provided, it is challenging to derive the actual organizational model (e.g. an organizational chart) in an organization. However, it is possible to recognize groups of resources that have similar characteristics relevant to the performance of a business process. For example, in [1] the authors propose a resource grouping mechanism based on how frequently the human resources carry out the same tasks, and suggest that the discovered organizational groups can be relevant to roles and functional units in which employees possess similar skills and knowledge to perform the tasks.

To date there have been a number of research efforts on mining organizational models from event logs (e.g. [1, 3, 4]), whereas almost all of these existing studies have made an assumption of *disjoint* organizational groups, which means that each resource is a member of *a single* organizational group. In fact, in many real-world organizations it is common to have employees who possess multiple skills to share roles and responsibilities across organizational groups. More generally, modern organizations emphasize the importance of having smooth and active communication among various functional units, and achieve so by setting up cross-department roles to enhance the coordination [5]. From the viewpoint of organizational structures, resources working across different organizational groups form the *overlap* between the groups. From the viewpoint of process improvement, such resources are likely to be the so-called *generalists* – a special category of resources that can help increase the flexibility of a business process [2]. In terms of flexibility, we consider the generalists to carry out a role similar to the role of "libero" in certain team sports.

In this paper we propose an approach for the discovery of organizational models from event logs, which allows the sharing of human resources between different organizational groups. By relaxing the assumption of disjoint organizational groups (applied in most of the existing work), new discovery algorithms are developed to address the challenges arising from dealing with the potential overlaps between organizational groups. Based on the characteristics of the problem of interest, a couple of existing cluster analysis techniques (from the field of data mining) are chosen and applied in our discovery algorithms. Experiments are conducted on an implementation of the discovery algorithms, using real-life event logs, to evaluate the applicability and effectiveness of our approach.

The contribution of our work is twofold. On the one hand, the discovered organizational model with potential overlaps is a better reflection of the actual organizational grouping of resources relevant to process execution, and hence it will enable more insightful resource performance analysis. On the other hand, identifying resources that belong to more than one organizational group from event logs presents a novel data-driven approach to the discovery of generalists in an organization and their organizational positioning (i.e. in which organizational groups they perform in practice). Finding the information about generalists will help improve resource utilization and also serve as an important step for actionable process improvement. For example, one strategy for process improvement is to keep such resources free when possible, which guarantees flexibility in the distribution of work [6].

The rest of the paper is organized as follows. Sect. 2 provides a review of the related work on the topic. Sect. 3 introduces basic concepts and preliminary notions. In Sect. 4, we present our approach for mining organizational models with overlaps, and in Sect. 5 we discuss the experiments and analyze the evaluation results. Finally, Sect. 6 concludes the paper and outlines future work.

2 Related Work

The research considering the organizational perspective of process mining originates from the work by van der Aalst et al. [7], in which several types of inter-resource relationship metrics are defined for deriving resource social networks from event logs. Based on the analysis of resource social networks, Song and van der Aalst [1] propose the conceptual framework of organizational mining as a sub-field of process mining, within which three research dimensions of organizational mining are proposed: *discovery*, *conformance checking* and *extension*.

Discovery refers to constructing models that reflect the reality. In the context of organizational mining, these models include organizational models, social networks and resource assignment/allocation rules. *Organizational model mining* focuses on finding the grouping of resources (employees), e.g. who belongs to which functional unit [1, 8], who plays what roles [3, 9] or holds what social positions in collaboration [10]. Recently, the work of Appice [8] introduces an approach for mining organizational models using a community detection technique, which makes no assumption about each resource belonging to a single group. To the best of our knowledge, this is so far the only existing approach capable of deriving organizational models with potential overlaps.

The discovery of social networks emphasizes the use of social network analysis to help understand the structure of communication between individual resources as well as between organizational groups [4, 7, 11]. The research presented in [12, 13] studies the discovery of rules related to staff assignment (who is allowed to do which tasks) and runtime activity distribution (to whom a specific task is allocated) to help with diagnosis and optimization of pre-defined rules.

In addition, there is also existing research concerning the organizational perspective of business processes at the level of individual resources. For example, in [14] the authors analyze the correlation between the workload of individual resources and their performance, and in [15] the authors propose a framework for analyzing and evaluating different resource behaviors in order to provide insights towards more informed resource-related decisions for performance improvement.

3 Preliminaries

Here we present several preliminary concepts necessary for describing the problem, following the conceptual framework of organizational mining defined by Song and van der Aalst [1]. A typical event log usually consists of a set of uniquely identifiable cases corresponding to the instances of an underlying business process. Each case contains a sequence of events that describe the activities carried out by some resources. Table 1 gives an example fragment of an event log recorded by a process-aware information system. Each row refers to one single event, which is described using attributes such as activity label, timestamp, and identity of the originating resource[1].

Table 1. An example fragment of an event log.

Case ID	Event ID	Activity label	Resource	Timestamp
c_1	e_1	Register request	John	2018/01/03 10:59:06
c_1	e_2	Examine thoroughly	Mike	2018/02/03 11:10:13
c_1	e_3	Decide	Clare	2018/02/21 15:43:32
c_1	e_4	Reject request	John	2018/02/22 10:35:52

Definition 1 (Event Log [7]). *Let T be a set of tasks and R be a set of resources. $E \subseteq T \times R$ is the set of events that denote the execution of tasks by originator resources. For any event $e \in E$, $\pi_t(e) \in T$ is the task being executed (or the activity) in e and $\pi_r(e) \in R$ is the originator resource of e. $C = E^*$ is the set of possible event sequences (traces describing a case). $L = \mathcal{B}(C)$ is an event log, where $\mathcal{B}(C)$ is the set of all bags (multi-sets) over C.*

In Definition 1 we do not take into account the ordering of events in a case. We focus on two standard attributes of an event – task and resource identity. We use them to build a simple "profile" for each resource, which reflects the history of the resource performing activities. Accordingly, a *performer by activity matrix* can be used to represent the profiles of a set of resources given an event log.

Definition 2 (Performer by Activity Matrix, adapted from [7]). *Given an event log L, let $\{e_1, ..., e_n\}$ be the set of all possible events recorded in L. The performer by activity matrix is an integer-valued matrix X of size $|R| \times |T|$, in which each row vector corresponds to the execution history of activities for a specific resource. Each element of X denotes the count of frequencies of a resource $r_i \in R$ conducting a specific task $t_j \in T$, defined as:*

$$X_{ij} = \Sigma_{1 \leqslant k \leqslant n} \begin{cases} 1, & \textit{if } \pi_r(e_k) = r_i \textit{ and } \pi_t(e_k) = t_j \\ 0, & \textit{otherwise} \end{cases}$$

where $1 \leqslant i \leqslant |R|$ and $1 \leqslant j \leqslant |T|$.

[1] For illustration purposes, resource name is used in the example in Table 1.

Simply consider the example fragment of an event log shown in Table 1. The performer by activity matrix build from this example based on Definition 2 is shown in Table 2. Below, we propose a generic and simple definition of an organizational group as a non-empty group of human resources (i.e. employees) in an organization. For each organizational group, we define a membership indicator associated with each resource to specify whether or not the resource belongs to the group.

Definition 3 (Organizational Group). *Let R be a set of (human) resources in an organization, an organizational group can be defined as $G \subseteq R$ and $G \neq \varnothing$. Given an organizational group G, for any $r \in R$, we define a membership indicator function $\mathbb{I}_G : R \to \{0, 1\}$ where $\mathbb{I}_G(r) = 1$ if $r \in G$ and 0 otherwise.*

Finally, we define the concept of organization model. It is simply considered as one entire group of several organizational groups defined in the above.

Definition 4 (Organizational Model). *An organizational model O is a set that consists of a finite number of (k) organizational groups $\{G_1, \ldots, G_k\}$. For any resource r that is part of the organizational model O, r belongs to one or more than one organizational group in O. That is, $\forall r \in \bigcup_{G \in O} G$, $\sum_{G \in O} \mathbb{I}_G(r) \geqslant 1$.*

As mentioned before, most of the existing studies in organizational mining apply the assumption of disjoint organizational groups in an organization, and hence they require that each resource should only belong to a single organizational group (i.e. $\forall r \in \bigcup_{G \in O} G$, $\sum_{G \in O} \mathbb{I}_G(r) = 1$). In Definition 4, our focus is to relax such assumption by recognizing that resources may belong to more than one organizational group in reality and thus to allow potential overlaps between different organizational groups.

4 Approach

Organizational model mining aims at recognizing groups of resources having similar characteristics. We concern the connection between this and the purpose of cluster analysis in data mining, which is to group a set of data objects into multiple clusters such that objects within a cluster have high similarity but are dissimilar to those in other clusters [16]. As a relatively mature field, there exist various types of techniques developed to provide solutions for different requirements and contexts. Since our intention is to derive results in which one resource may be member of more than a single organizational group, we select the technique of *overlapping clustering*, which allows flexible assignment of one data object to multiple clusters. In this paper, we design an approach adopting the idea of overlapping clustering to solve the problem of discovering organizational model with overlaps. Fig. 1 gives an overview of the three-phased procedure. We start from constructing the performer by activity matrix that characterizes the resources. Then we transfer the problem into cluster analysis and apply the selected model and algorithm to produce the clustering result, from which we derive an organizational model as the end result.

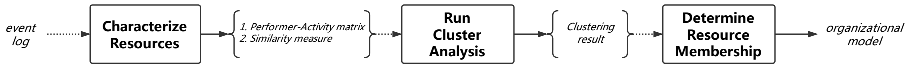

Fig. 1. The designed procedure for discovering organizational model with overlaps.

4.1 Characterizing Resources

Given an event log, we construct the performer by activity matrix by directly following Definition 2 and determine the execution frequencies while iterating over the events. Table 2 shows the result of deriving the matrix using the example event log fragment in Table 1 as input.

Table 2. The performer by activity matrix built from the example event log fragment.

	Activity 1	Activity 2	Activity 3	Activity 4
	Register request	Examine thoroughly	Decide	Reject request
John	1	0	0	1
Mike	0	1	0	0
Clare	0	0	1	0

Once the performer by activity matrix has been built, we need to select a measure for quantifying the similarity between any two resources by comparing the corresponding row vectors, in order to further group similar resources and derive an organizational model. Some variants of distance-based metrics provide meaningful measures in a process mining context. The Hamming distance, for example, accounts for whether or not two resources have executed the same types of tasks. Meanwhile, correlation-based metrics such as Pearson's correlation coefficient provide a view of statistical correlation. The choice of similarity measure should be done depending on the purpose and context of analysis.

For the next step, we apply the clustering techniques in order to obtain the clusters of resources. Two possible solutions are presented then. Since these two vary in terms of the deciding the final clusters, we will describe how to derive the end result, i.e. the output organizational model, respectively.

4.2 Solution 1: Cluster Analysis using a Mixture Model

We first elaborate on how to correlate the current problem with the concepts of overlapping clustering. The concept of probabilistic cluster and the hypothesis of mixture models are commonly used in cluster analysis to characterize the flexible assignment of one object to multiple clusters simultaneously. The hypothesis states that the latent categories hidden in the data objects could be mathematically represented using a series of distribution functions [16]. Each data object is related to each latent category by a sampling probability, and is viewed as a sample drawn from a mixture of distributions. In the context of our problem, we can regard the execution history of activities (i.e. the row vector in the performer by activity matrix corresponding to a resource) as the result of a resource following the work patterns of the organizational group(s) it belongs to. If the resource is indeed a member of several different groups, then its execution history of activities should be the consequence of multiple work patterns. We may therefore adopt the hypothesis of mixture models as an idea for a solution. First, cluster the resources by leveraging the performer by activity matrix along with the specified similarity measure and find the distribution function for each cluster, then for each row vector we calculate a sampling probability related with each cluster, which could be used to decide the membership of the resource.

Following the idea we could apply a classic Gaussian mixture model (GMM) as the first solution. In GMM we assume a Gaussian distribution for each latent category, and apply the well-founded EM algorithm [16] to fit the mixture model using the performer by activity matrix. EM works in an iterative fitting process, which starts with a random initialization and updates the mixture model greedily towards a higher value of the goal function (the likelihood of sampling all the vectors using the current model). The mixture model converges as the goal function value no longer increases or updates by a very trivial scale.

Using the converged mixture model, we can calculate the posterior probability of a row vector relating with each cluster, and take the result as the sampling probability. However, for actually deciding the membership of a resource, we need to choose a threshold to be applied on the probability value, which determines if the resource belongs to one or several of the groups. For example, if the chosen threshold value is 0.5, then the resource should belong to a group only if its related sampling probability is larger or equal to 0.5.

For the basic solution using GMM, we notice some problems related to its configuration. Before starting the fitting process, it requires us to decide the number of clusters upfront. This should be done based on the control of granularity we desire: with a higher number it enables us to discover more fine-grained groups, which may be the very specific roles or small workgroups, whereas a lower number of clusters would possibly lead to finding departments at a higher level. Another problem concerns the thresholding step applied for the purpose of deciding resource membership. It is hard to determine an effective level of probability value that decides whether a resource *indeed* belongs to an organizational group or not: for instance, for a fitted GMM we could calculate the result that an involved employee Jack has the probability value of 0.49 that he belongs to Group 1, and 0.51 that he belongs to Group 2. The question is: how should we actually decide Jack's membership given these numbers? Selecting an appropriate threshold value may become a challenging task, since the scale of the estimated posterior probabilities lack a solid interpretation in the context of organizational model mining. We therefore present another overlapping clustering algorithm that addresses the challenge.

4.3 Solution 2: Cluster Analysis using a More Generative Model

Consider the example of deciding Jack's membership illustrated before. The use of mixture models like GMM poses the challenge of configuring proper threshold parameter, which may hinder us from directly applying the method for discovering organizational models. The challenge arises from the underlying hypothesis of mixture models: when we view the row vector corresponding to a resource as a data object being clustered, the posterior probabilities that we use for later deriving membership only indicate the possibilities of having the current data object sampled from each of the distributions *independently* [16]. Hence a mixture model may fail in well characterizing the reality that, for resources with multiple memberships across several groups, their execution history of activities results from the *joint* effect of all the work patterns of the groups.

Without shifting from the general concepts of both organizational model mining and overlapping clustering, we seek to find a more natural and descriptive model that avoids deriving membership from probabilities, and constitutes a better solution for the current problem.

The Model-based Overlapping Clustering (MOC) model [17] bases itself on the same concepts of probabilistic clusters as GMM does, but without employing the hypothesis of having objects sampled from a mixture of distributions related with the latent categories. Instead, a boolean-valued membership vector is defined directly for each of the objects to be clustered, of which the values are inferred after fitting the model with the data. In comparison with mixture models, the MOC model is a more natural generative model for overlapping clustering. In MOC the data objects being clustered are hypothesized to be generated by simultaneously considering multiple components, as each of the components refers to a part of the model that relates to one of the latent categories to be discovered (similar to the distribution functions).

Algorithm 1 depicts the procedure of applying the MOC model to the current problem. We omit some of the mathematical details here for brevity, for which one may refer to [17] for a more in-depth explanation. Given n resources and the related event log, we assume that the performer by activity matrix X and similarity measure have been decided prior to running the algorithm, and the granularity of analysis has been specified already, i. e. k groups to be discovered. The algorithm starts by an initial estimate of the membership matrix M, which is usually initialized in a random manner. Another model parameter to be initialized is a matrix that represents the active status of each component in the MOC model, denoted as A, for which random initialization will be fine.

After the initialization of the model parameters we proceed to the iterative process for fitting the model to the data (Line 3-10). At each iteration we first update the value of A directly using the current M and X [17]. In the next step, for each membership vector M_i we try to find a value that maximizes the metric value. The search may be time-consuming when the desired group number k is large, however certain algorithms could be plugged in here to speed up the search process [17]. When the appropriate setting of M has been obtained, we calculate the value of the goal function defined here as the log-likelihood (Line 7), and compute the increase in comparison to the result of the last iteration. The iterative updating process stops when convergence is reached, i.e. the increase in the goal function value is sufficiently small.

Algorithm 1: Applying MOC for Discovering an Organizational Model

```
   Input:
     – $\{r_1, \ldots, r_n\}$: the $n$ resources involved;
     – $X$: the constructed performer by activity matrix, also assuming that the
       similarity measure has been specified accordingly;
     – $k$: the number of organizational groups expected to be discovered
       (depending on the desired granularity).
   Output: $O$: the resulting organizational model consisting of $k$ groups.
   // Step 1: Initialize the membership parameter
 1 Initialize an $n \times k$ boolean value matrix $M$, where each of the $n$ row vectors
     indicates the membership of a corresponding resource
 2 Initialize a $k \times d$ real value matrix $A$ that denotes the active status of each
     component in the model
   // Step 2: Fit the model to the data through iterative updating
      until convergence
 3 repeat
       // Update $A$ by direct computing the value from $X$ and $M$
          (cf. [17])
 4     $A \leftarrow \mathit{update}\,(A, X, M)$
       // Update $M$ by searching a setting that maximizes the selected
          similarity measure
 5     for $i = 1$ to $n$ do
 6         $M_i \leftarrow \underset{M_i \in \{0,1\}^k}{\text{argmax}}\ \text{SIMILARITY\_MEASURE}\,(X_i, M_i A)$
 7     end
       // Calculate the goal function value using the log-likelihood
          (cf. [17])
 8     $L = \log P\,(X, M, A)$
 9     Calculate the increase $\Delta L$ by comparing with the last iteration
10 until $\Delta L$ is sufficiently small
   // Step 3: Derive the resulting organizational model utilizing the
      membership matrix
11 Initialize $k$ empty sets $G_1, G_2, \ldots, G_k$
12 for $i = 1$ to $n$ do
13     for $j = 1$ to $k$ do
14         if $M_{ij} = \mathit{true}$ then
15             $G_j \leftarrow G_j \cup \{r_i\}$
16         end
17     end
18 end
19 return $O = \{G_1, G_2, \ldots, G_k\}$
```

With the fitted model we can now derive the end result in a straightforward way, since the membership of all the n resources has been determined as the value of the $n \times k$ membership matrix M. Therefore, we just need to simply assign the resources to the corresponding ones of the k sets (Line 11-18), and return the resulting sets as the discovered organizational groups.

Comparing to the more naïve solution of GMM, the solution using MOC model avoids introducing probabilities as the degree of resource membership, and therefore addresses the challenge of having to select thresholds. Given the event log and resources to be analyzed, users would only need to focus on the resource profiling phase, and then set up the expected number of groups. The end result will be an organizational model containing the exact number of groups as required, where overlaps are allowed to exist.

5 Evaluation

5.1 Experiment Design

Both solutions (applying either GMM or MOC) have been implemented in a standalone demo[1]. We evaluated their feasibility on real-life event log data. We aim at giving empirical validation on whether the proposed solutions work effectively in discovering organizational models when there indeed exist overlaps among organizational groups.

Event Logs. Different from the evaluation methods in the previous research on the problem (cf. [1], [8], [11]), the purpose of the validation here requires us to be aware of the "ground truth" information relevant to the internal groups in an organization a priori. For this purpose we picked two sets of real-life event logs, namely "WABO" and "Volvo". The background of these event log datasets are as follows:

- WABO: The event log from the WABO dataset contains the records of the receiving phase of an environmental permit application process in an anonymous municipality within the CoSeLoG project [18].
- Volvo: This dataset includes event logs generated from the problem management system VINST of Volvo Belgium, which was originally released for the BPI Challenge 2013 [19]. It contains the event logs that describe several business processes handling incidents and problems in the IT-services delivered and/or operated by Volvo IT. We choose the event log related with the process managing the open problems for experiment use.

The event logs are recorded in the IEEE standard XES format [20], and include an extended event attribute termed `org:group`, which indicates the group identity of the resource that triggered the event. We recognize that the ground truth organizational models can be extracted by utilizing this information of identities, which can then serve as the reference models for our experiments. To do this we first filter out the events with missing values on `org:group` (including both null and invalid ones). Then we extract the ground truth organizational model by putting resources together into groups accordingly, based on the `org:group` values they relate to as event originators. Table 3 gives a brief

[1] https://github.com/royyjing/bpm-2018-Yang_Find

overview of the preprocessed event logs, along with some basic statistics of the extracted reference models: the average size of groups (Avg. group size), and the average number of groups that a resource belongs to (Avg. membership). One may recognize immediately the existence of overlaps in the reference models after inspecting the basic statistics shown in the table. A further comparison on Avg. membership reveals that the overlapping condition is less obvious in the Volvo case (Avg. membership 1.176 while WABO has a value of 3.886), suggesting considerably fewer employee resources possessing multiple group identities in Volvo IT.

Table 3. Overview of the event logs and the extracted reference models.

Event log	Cases	Events	Activities	Resources	Organizational groups	Avg. group size	Avg. membership
WABO	1,348	6,641	27	44	**9**	**19.0**	**3.886**
Volvo	818	2,331	5	239	**11**	**25.5**	**1.176**

Experiment Setups. We conducted the experiments using the comprison method. Two methods proposed in previous research are selected as baseline: a traditional partitioning method that produces disjoint organizational models [1], namely MJA; and a community detection based method developed by Appice [8] that is capable of deriving organizational models with possible overlaps, namely Commu. We examine if the organizational models discovered from the same source of event logs using GMM and MOC can better capture the reality, i.e. more similar to the reference models.

To start with, we build the performer by activity matrix, and choose the Pearson's correlation coefficient as the metric for similarity measure. Since the setup of the algorithms involved in evaluation may vary, we decided to configure the parameters for each algorithm separately, as long as they produce resulting organizational models with exactly the same number of organizational groups discovered as that in the reference ground truth.

Evaluation Metrics. For the purpose of comparing between the results of discovery and the reference models to assess the effectiveness of different methods, we consider adopting extrinsic evaluation metrics. One example is the entropy measure [1], which can be used for measuring the scale of difference between a generated model and the referenced one. However, as the current research has been extended to the overlapping situation, the entropy measure becomes inappropriate as well as many other commonly used extrinsic measures. We therefore turn to the extended BCubed metrics (including BCubed Precision, Recall and F-measure) [21], as they are applicable for evaluation on the overlapping cases. From an organizational model mining point of view, the meaning of the BCubed metrics can be interpreted as follows:

1. BCubed Precision represents the ratio of how many resources in a same discovered organizational groups belong to the same actual groups. A higher value of BCubed Precision means fewer mistaken assignments in the discovered organizational model.
2. BCubed Recall represents the ratio of how many resources from a same actual groups are assigned to the same discovered organizational groups. A higher value of BCubed Recall means more resources with the same actual group identities are placed together by the mining algorithm.
3. BCubed F-measure is a combination of BCubed Precision and Recall, defined as the harmonic average of the two.

Besides the BCubed metrics, we also want to compare the basic statistics of the discovered organizational model (Avg. group size and Avg. membership), with those of the groundtruth model.

5.2 Comparing with the Disjoint Partitioning Method

In the first experiment we wish to compare our solutions with the disjoint partitioning method MJA. The idea behind MJA is to view the resources as vertices in a graph, and connect weighted edges between them based on the measured similarity values. By eliminating certain edges by a threshold value, the original graph is further partitioned into several connected components, which are taken as organizational groups that constitute the final organizational model. The result generated from MJA is obviously disjoint.

Table 4 shows the evaluation results measured by the BCubed metrics. From the table we can see that MJA obtains higher precision rates. However, the disjoint nature of MJA prevents it from recognizing the fact that similar resources may possibly share more than one group identities in an overlapping organizational model. Thus, for MJA, similar resources are clustered into one group only, which lead to the relatively lower recall.

On the other hand, the proposed solutions using either GMM or MOC have comparatively lower precision yet higher recall values. It can be explained that both overlapping clustering based algorithms tend to put more resources into the groups, which is consistent with the larger group sizes shown in Table 5. This leads to the better recall rates, but at the same time makes the discovered organizational groups contain relatively members being mistakenly assigned, which directly cause the lower precision of GMM and MOC.

Moreover, for the Volvo case we notice that even the baseline MJA produces a relatively lower precision, and the situation of recall rates mentioned above becomes even more significant. The reason is due to the large total number of resources compared to the much smaller number of activity types (239 resources compared to 5 activity types). The smaller number of activity types leads to fewer columns in the performer by activity matrix, and may therefore weaken the effect of measuring similarity.

Despite the observation that GMM and MOC may tend to sacrifice some precision rate and bring mistaken assignments, from Table 5 we can draw a conclusion – the overlapping-clustering-based solutions are able to derive an overlapping organizational model that captures the reality, whereas methods like MJA holding the assumption of disjoint organizational model are not.

Nevertheless, we still have the following questions: How effective are our solutions comparing to other solutions that can also produce overlapping organizational models? Will the other solutions also encounter the problem of unsatisfying precision? We will explore the answers to these questions through the following experiment and analysis.

Table 4. Results of comparing with MJA on the BCubed metrics.

Event log	BCubed Precision			BCubed Recall			BCubed F-measure		
	MJA	GMM	MOC	MJA	GMM	MOC	MJA	GMM	MOC
WABO	**0.814**	0.624	0.757	0.213	**0.812**	0.735	0.337	0.706	**0.745**
Volvo	**0.496**	0.186	0.24	0.397	**0.944**	0.94	**0.441**	0.31	0.382

Table 5. Results of comparing with MJA on the grouping statistics.

Event Log	Avg. group size				Avg. number of membership			
	Ground truth	MJA	GMM	MOC	Ground truth	MJA	GMM	MOC
WABO	19.0	4.9	28.4	22.8	3.886	**1**	5.818	4.659
Volvo	25.5	21.7	146.5	110.9	1.176	**1**	6.745	5.105

5.3 Comparing with the Community-Detection Based Method

In this experiment we choose as baseline a community detection based approach [8] which we refer to as Commu. Our goal is to make a comparison between the effectiveness of Commu and our approach. Commu is based on social network analysis techniques rather than cluster analysis, but shares the same purpose of grouping cohesive resources into communities that represent the internal organizational groups. It applies the linear network model with the Louvain algorithm, and derives organizational models which allow the existence of overlapping communities (organizational groups).

Tables 6 and 7 show the evaluation results of this experiment. By observing the average number of membership we first confirm that the baseline method Commu indeed generates overlapping results. For the BCubed metrics, we notice that GMM performs roughly the same as Commu, whereas MOC performs better than Commu in both cases. And the grouping statistics show that the models produced by using either GMM or MOC are more realistic compared with Commu.

Meanwhile, we learn from the tables that Commu also produced a result of low precision and oversize groups, as in the Volvo case, and even worse while comparing with GMM and MOC (refer to the grouping statistics in Table 7).

In general, we may conclude that our approach is more effective as a solution to discovering organizational models with overlaps, compared to the community detection based method. Nevertheless, as both methods have the shortcoming of introducing mistaken assignment of resources to groups causing low precision and unrealistic group sizes, further work is needed to address this shortcoming.

Table 6. Results of comparing with Commu on the BCubed metrics.

Event log	BCubed Precision			BCubed Recall			BCubed F-measure		
	Commu	GMM	MOC	Commu	GMM	MOC	Commu	GMM	MOC
WABO	0.718	0.624	**0.757**	0.651	**0.812**	0.735	0.683	0.706	**0.745**
Volvo	0.195	0.186	**0.24**	**0.948**	0.944	0.94	0.324	0.31	**0.382**

Table 7. Results of comparing with Commu on the grouping statistics.

Event Log	Avg. group size				Avg. number of membership			
	Groundtruth	Commu	GMM	MOC	Groundtruth	Commu	GMM	MOC
WABO	19.0	28.8	28.4	**22.8**	3.886	5.886	5.818	**4.659**
Volvo	25.5	152.6	146.5	**110.9**	1.176	7.025	6.745	**5.105**

5.4 Discussion

We can draw some interesting insights considering results from both experiments conducted. The first conclusion concerns the comparison of effectiveness between GMM and MOC. It has been evaluated through the experiments that MOC performs better, indicated by the higher precision and F-measure, along with the grouping characteristics being more similar to the ground truth model. Taking into consideration that it requires no cumbersome decision to set up the extra threshold parameter when applying MOC, we conclude that MOC will serve as a better solution than GMM for discovering organizational models with overlaps.

On the other hand, we also realize that for our solution, there exists a shortcoming which would become significant when the latent organizational model is less overlapped. We infer the possible reasons behind it as twofold. The first one concerns the relatively fewer types of activities compared to the number of resources. The second concerns the lack of constraints on the number of groups allowed for each resource to be assigned to. As the former is limited by the content of the event log, we discuss the remedy for the latter.

Given no constraints, both GMM and MOC may try to relate resources to many organizational groups as long as the goal function value is being optimized.

This eventually causes the unrealistic mining result in which one resource is a member of considerably many organizational groups simultaneously, diverging from the reality that some resources may possess few or no shared group identities, as in the Volvo case. To solve this, a natural idea is to set up the constraints to mitigate the problem of involving too many resources. Yet this would require more prior knowledge of the underlying organizational structure to implement. Nevertheless, we argue that such an improvement needs only slight modification on the current solution. For GMM, it requires the proper threshold value. For MOC, heuristics are to be introduced to prune the search space in updating the estimate of membership. Another remedy could be mixing application of the proposed solution with the traditional disjoint method: Given an organizational model mining task with the performer by activity matrix has been built along with the specified similarity measure, one may first mine a disjoint model using the traditional method, and utilize the obtained model statistics for the guided initialization of the parameters. Then, apply GMM or MOC to discover an organizational model with potential overlaps. We plan to leave the exploration for improvement to our future research on the topic.

6 Conclusion

Organizational model mining techniques enable the discovery of organizational models from event logs. In this paper, we relax the assumption of disjoint organizational groups held by existing methods and discover organizational models in which individual resources may share multiple group identities. We refer to overlapping clustering techniques and introduce two solutions, GMM and MOC, for deriving organizational models with overlaps. Results from experiments on real-life event log data demonstrate the applicability and effectiveness of the methods. We also recognize the potential limitation of our solution and conclude the reasons behind it, which lead to identifying the potential heuristics for further amending the current approach.

In future work we will consider the following aspects: (1) to improve our approach by effectively incorporating the identified heuristics; (2) to link the current research with performance analysis on generalist resources; (3) to conduct evaluation on more real-life cases.

Acknowledgments. This work is supported by the National Key Research and Development Program of China (Grant No. 2017YFB0202200); the National Natural Science Foundation of China (Grant No. 61572539); the Research Foundation of Science and Technology Plan Project in Guangdong Province (Grant No. 2016B050502006); and the Research Foundation of Science and Technology Plan Project in Guangzhou City (Grants No. 2016201604030001, 201704020092).

References

1. Song, M., van der Aalst, W.M.P.: Towards comprehensive support for organizational mining. Decision Support Systems **46**(1) (2008) 300–317

2. Reijers, H., Mansar, S.L.: Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. Omega **33**(4) (2005) 283 – 306
3. Jin, T., Wang, J., Wen, L.: Organizational modeling from event logs. In: International Conference on Grid and Cooperative Computing (GCC). (2007) 670–675
4. van Zelst, S.J., van Dongen, B.F., van der Aalst, W.M.P.: Online discovery of cooperative structures in business processes. In: OTM Confederated International Conferences, Springer (2016) 210–228
5. Daft, R.L.: Organization Theory and Design. (2010)
6. van der Aalst, W.M.P., van Hee, K.: Workflow Management: Models, Methods, and Systems. MIT Press, Cambridge, MA, USA (2004)
7. van der Aalst, W.M.P., Reijers, H.A., Song, M.: Discovering social networks from event logs. Computer Supported Cooperative Work (CSCW) **14**(6) (2005) 549–593
8. Appice, A.: Towards mining the organizational structure of a dynamic event scenario. Journal of Intelligent Information Systems **50**(1) (Feb 2018) 165–193
9. Burattin, A., Sperduti, A., Veluscek, M.: Business models enhancement through discovery of roles. In: IEEE Symposium on Computational Intelligence and Data Mining (CIDM). (2013) 103–110
10. Liu, R., Agarwal, S., Sindhgatta, R.R., Lee, J.: Accelerating collaboration in task assignment using a socially enhanced resource model. In: Business Process Management, Springer (2013) 251–258
11. Ferreira, D.R., Alves, C.: Discovering user communities in large event logs. In: International Conference on Business Process Management. (2011) 123–134
12. Rinderle-ma, S., van der Aalst, W.M.P.: Life-cycle support for staff assignment rules in process-aware information systems. Technical Report 213, TU Eindhoven (2007)
13. Schönig, S., Cabanillas, C., Jablonski, S., Mendling, J.: A framework for efficiently mining the organisational perspective of business processes. Decision Support Systems **89** (2016) 87 – 97
14. Nakatumba, J., van der Aalst, W.M.P.: Analyzing resource behavior using process mining. In: International Conference on Business Process Management, Springer (2009) 69–80
15. Pika, A., Leyer, M., Wynn, M.T., Fidge, C.J., ter Hofstede, A.H.M., van der Aalst, W.M.P.: Mining resource profiles from event logs. ACM Trans. Manage. Inf. Syst. **8**(1) (March 2017) 1:1–1:30
16. Han, J., Pei, J., Kamber, M.: Data mining: concepts and techniques. Elsevier (2011)
17. Banerjee, A., Krumpelman, C., Ghosh, J., Basu, S., Mooney, R.J.: Model-based overlapping clustering. In: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. (2005) 532–537
18. Buijs, J.: Receipt phase of an environmental permit application process (WABO), CoSeLoG project (2014)
19. Steeman, W.: BPI challenge 2013 (2013)
20. IEEE: IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. Technical report (Nov 2016) IEEE Std 1849-2016.
21. Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Information Retrieval **12**(4) (2009) 461–486