1-1-2010

Business rules discovery from process design repositories

Jantima Polpinij
University of Wollongong, jp989@uow.edu.au

Aditya K. Ghose
University of Wollongong, aditya@uow.edu.au

Hoa Khanh Dam
University of Wollongong, hoa@uow.edu.au

Recommended Citation
Polpinij, Jantima; Ghose, Aditya K.; and Dam, Hoa Khanh: Business rules discovery from process design repositories 2010, 614-620.
https://ro.uow.edu.au/infopapers/1724

Business rules discovery from process design repositories

Abstract

Traditional process mining approaches focus on extracting process constraints or business rules from repositories of process instances. In this context, process designs or process models tend to be overlooked although they contain information that are valuable for the process of discovering business rules. This paper will propose an alternative approach to process mining in terms of using process designs as the mining resources. We propose a number of techniques for extracting business rules from repositories of business process designs or models, leveraging the well-known Apriori algorithm. Such business rules are then used as a prior knowledge for further analysing, verifying, and modifying process designs.

Keywords

Business, rules, discovery, from, process, design, repositories

Disciplines

Physical Sciences and Mathematics

Publication Details

Polpinij, J., Ghose, A. K. & Dam, H. Khanh. (2010). Business rules discovery from process design repositories. Proceedings: 2010 6th World Congress on Services (Services-1) (pp. 614-620). Piscataway, New Jersey, USA: IEEE.

2010 IEEE 6th World Congress on Services

Business Rules Discovery from Process Design Repositories

Jantima Polpinij Aditya K. Ghose Hoa Khanh Dam
Decision Systems Lab (DSL)
School of Computer Science and Software Engineering (SCSSE)
Faculty of Informatics, University of Wollongong
Wollongong 2500 NSW Australia
{jp989, aditya, hoa}@uow.edu.au

Abstract— **Traditional process mining approaches focus on extracting process constraints or business rules from repositories of process instances. In this context, process designs or process models tend to be overlooked although they contain information that are valuable for the process of discovering business rules. This paper will propose an alternative approach to process mining in terms of using process designs as the mining resources. We propose a number of techniques for extracting business rules from repositories of business process designs or models, leveraging the well-known Apriori algorithm. Such business rules are then used as a prior knowledge for further analysing, verifying, and modifying process designs.**

Keywords: Business Rules, Process Designs, the Apriori Algorithm

I. INTRODUCTION

A business process [1, 2] is a group of logically related tasks that use the enterprise's resources to meet customer demands and achieve the organization's objectives. Documenting and managing business processes can play an important role in achieving efficiency within an enterprise. In general, business processes can be designed and represented using graphical notations such as flow charts, functional flow block diagrams, data flow diagrams, control flow diagrams, Gantt charts, PERT diagrams, BPMN and so on [3]. In this context, business process management involves defining business processes, defining their goals and specifying how they fit with other business processes [1, 2, 3].

Business process designs or models[1] are often kept in a repository, which can be used later to support process redesign. There are many drivers for business process redesign [4, 5]. These include process compliance, process re-purposing/re-use and process optimization. In fact, the use of previous information such as process designs may enable organizations to quickly respond to changes within the business environments.

On the one hand, it may be difficult to reuse such valuable information if the users are not the persons who produced them. On the other hand, in order to develop high-quality business processes, the business process designer should have adequate understanding of the relevant business rules. This would increase the cost and effort for analyzing and designing business process.

Therefore, a number of approaches have been proposed to support business process design in terms of extracting business rules from existing resources. In the last decade, data mining has been used as a tool for knowledge management in organizations. Data mining refers to the extraction of knowledge from large data sets though identification of patterns within the data [6]. Data mining practices have been developed and adapted to understand a business process and other process-related models installed within an organization [7]. In addition, it can mine the needs of an organization to reconstruct actual business processes.

A number of attempts have been made to mine processes but they have relied on event logs recorded by IT systems. In particularly, process mining [8, 9] was proposed to provide the capability to discover, detect, control, organize, and monitor actual process execution by extracting useful knowledge from event logs (i.e. process instances). This class of techniques is often used when no formal description of the process can be obtained by other means, or when the quality of existing documentation is poor. In addition, it is possible to use process mining to monitor deviations (by comparing event logs with process designs to determine whether execution scenarios conformed to normative process designs. However, there are settings where process mining proves to be inadequate. Firstly, if a process has never been performed, no event logs information is available, rendering the technique useless. Secondly, learning/data mining techniques such as those used in this approach to process only return reliable results when there are sizeable data-sets (in the case of process mining, event logs) available. If the size of the set of event logs is small (which is often the case with infrequently executed processes, such as those associated with disaster management, or with Greenfield applications), we cannot guarantee the reliability of the knowledge extracted.

Our work is motivated by a key driver. We aim to explore an alternative dimension to process mining where the

[1] Hereafter, business designs and business models are used interchangingly.

978-0-7695-4129-7/10 $26.00 © 2010 IEEE
DOI 10.1109/SERVICES.2010.73

objective is to extract process constraints or business rules [10], which can be a foundation for understanding the constraints on the core functions of a system, as opposed to process designs. We also focus on an alternative dataset – process designs as opposed to process instances. This offers the following benefits:

- If we can identify a set of key activities which are important for driving in processes success as a set of business rules, this can lead to efficiencies in various ways such as reducing time of verification and improvement of process designs. It seems that we are ranking a priority of activities in process. Therefore, when a process faces a problem, firstly, process designers may need to check the activities which are in a set of business rules. This is because these activities may lead a process broken and then it may become a cause of system failure.
- Also, there is another offer for this approach. Our business rules can be used to classify process designs. Process designs having the same business rules will be grouped in the same class. Slimming down the number of process designs contributes to a better analyzing and verifying of the impact of the process designs.

The key contribution of this paper is to provide a technique for extracting business rules from repositories of business process designs, leveraging the well-known Apriori algorithm [11]. Such business rules are then used as a prior knowledge for further analysing, verifying, and modifying process designs.

The remainder of the paper is organized as follows. We describe our methodology in Section 2 and an example of our approach is presented in Section 3. Experimental results are described in Section 4 along with the performance of our concepts. Finally, we summarize our research and discuss some future work directions in Section 5.

II. Business Rules Discover by The Apriori Algorithm

Our approach consists of two main steps which are described in this section.

A. *Pre-processing Process Designs*

In this step, we pre-process a collection of business process designs in order to prepare them for the next step - extracting a priori knowledge using the Apriori algorithm [11]. It is noted that directly extracting knowledge from business process designs represented on graphical diagram is a difficult task. Therefore, the collection of business process designs will be transformed into a textual format from which is easier to extract knowledge. Our approach uses the XML Metadata Interchange (XMI[2]) standard.

However, gateways within process designs make the application of the Apriori algorithm difficult. This is because it is possible that there are many alternatives following with a gateway condition. Thus, this is hard to analyze using the Apriori algorithm because its knowledge background is a large set of itemsets. Each *itemset* should be in a pattern of linear sequence [12, 13]. In order to ease the discovery of associations between activities in a set of processes, we transform business processes containing gateway(s) into a collection of linear processes. The number of sub-processes will depend on the decision types. For example, in BPMN (Business Process Modeling Notation)[3], the condition gateways can be the forking, merging, or joining of paths. In this case, any sub-process that is split from the original business process is equal to a business process. In fact, each sub-process is in the simple format without any condition. We are only interested the associations of activities in a business process collection. An example can be illustrated in Figure 1.

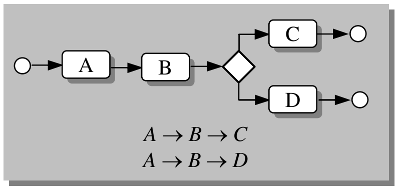

A
B
C
D
$A \rightarrow B \rightarrow C$
$A \rightarrow B \rightarrow D$

Figure 1. This process contains a condition gateway (◊). In order to support the stage of the Apriori algorithm analysis, the process will be spited into two sub-processes: $\{A \rightarrow B \rightarrow C\}$ and $\{A \rightarrow B \rightarrow D\}$.

B. *Extracting Business Rules*

In our work, business rules are extracted from a collection of business process designs using the Apriori algorithm. Traditionally, the Apriori Association rule mining that is defined by Agrawal et al [11] extracts interesting associations and/or relationships among large set of data items. This algorithm shows attribute value conditions that occur frequently together in a given dataset. The dataset is used to generate the desired rules.

In our work, the problem of the Apriori Association rule mining can be defined as follows: Let $A = \{a_1, a_2,\ldots, a_n\}$ be a set of *activities*. Let $P = \{p_1, p_2, \ldots, p_m\}$ be a set of processes called the *process collection*. Each process in P contains a subset of the activities in A called *activitysets* (in the original algorithm, a set of items is called *itemsets*). An *association rule* is defined as an implication of the form $X \Rightarrow Y$ where X, $Y \subseteq A$ and $X \cap Y = \emptyset$. X and Y are called *antecedent* and *consequent* of the rule respectively. To illustrate the concept of the Apriori Association rule mining, we use a small example of a business process collection (See in Figure 2).

[2] http://www.omg.org/technology/documents/formal/xmi.htm

[3] http://www.bpmn.org/)

The example has ten activities and four business process designs. Note that, after step 1, the processes 3 and 4 should be transformed into the simplified format (as shown in Figure 3). In this simplified format, process 3 reduces to two sequences of activities: ($a_5 \rightarrow a_6 \rightarrow a_7$) and ($a_5 \rightarrow a_6 \rightarrow a_8$). Similarly, process 4 is reduced as shown in Figure 3.

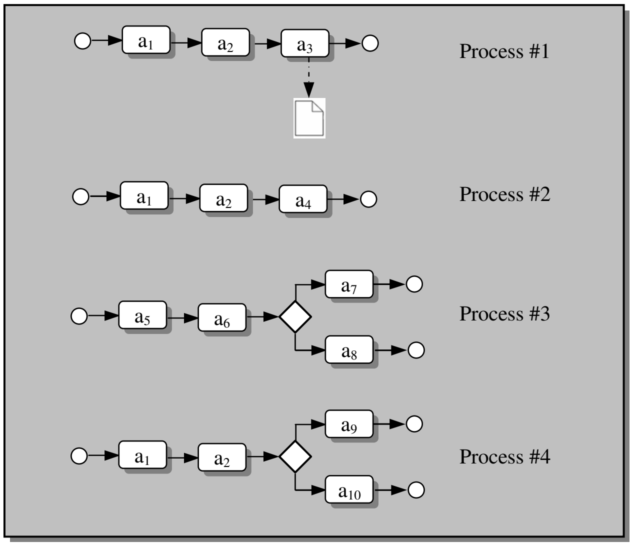


Figure 2. An example of a business process collection

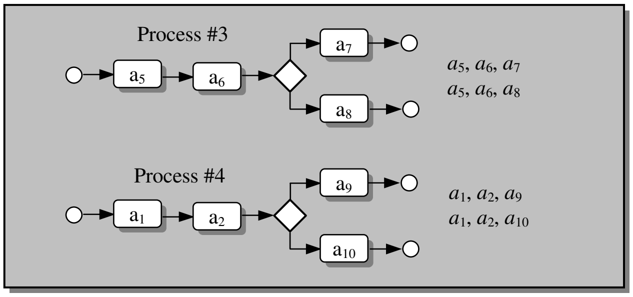
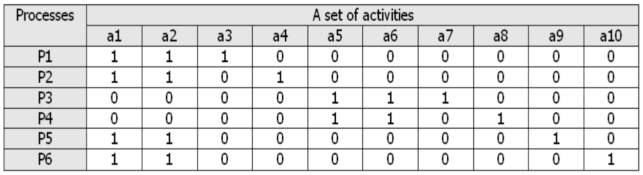


Figure 3. The examples of a business processes 3 and 4 which are transformed into a simple format in order to be appropriate in the stage of analyzing by the Apriori algorithm.

After step 1, these are ten activities and six business process designs[4]. Each business process and its activities are be shown in Table 1.

An example rule for this collection could be $\{a_1, a_2\}\rightarrow\{a_3\}$ meaning that if a_1 and a_2 are presented in a process, then a_3 should be presented in the process.

TABLE I. BUSINESS PROCESS AND THEIR ACTIVIES



Processes	A set of activities									
	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10
P1	1	1	1	0	0	0	0	0	0	0
P2	1	1	0	1	0	0	0	0	0	0
P3	0	0	0	0	1	1	1	0	0	0
P4	0	0	0	0	1	1	0	1	0	0
P5	1	1	0	0	0	0	0	0	1	0
P6	1	1	0	0	0	0	0	0	0	1

Note: '*1*' codes presence and '*0*' codes absence of an activity in a process.

Fundamentally, to select interesting rules from the set of all possible rules, this technique needs two parameters as quality measurements: the *Support Threshold* and the *Confident Threshold*. The framework is known as the *support-confidence framework* for association rule mining.

Let *P* be processes. The *support* of an activityset *X* is defined as the proportion of business processes in the collection which contain the activityset.

$$Support\ (X \Rightarrow Y) = \ Support\ (X \cup Y)$$

$$\frac{\#\ P\ containing\ X\ followed\ with\ Y}{\#\ P\ in\ a\ collection} \qquad (1)$$

As can be seen from Table 1, the example of activitysets $\{a_1, a_2\}$ has a support of 4/6 = 0.6666 since it occurs in 66.66% of all business processes.

Meanwhile, the confidence is given a rule in the form of "if *X* then *Y*". Rule confidence is defined as the conditional probability that *Y* is true when *X* is known to be true for a random instance:

$$Confidence\ (X \Rightarrow Y) = \frac{Support\ (X \cup Y)}{Support\ (X)}$$

$$\frac{\#\ P\ containing\ X\ followed\ with\ Y}{\#\ P\ containing\ X} \qquad (2)$$

In Table 1, the rule $\{a_1, a_2\}\rightarrow\{a_3\}$ has a confidence of 0.1666/0.6666 = 0.2499 in the collection, which means that 25% of the business processes containing a_1 and a_2 will be followed with a_3. Also, *confidence* can be interpreted as an estimate of the probability $P(Y|X)$, the probability of finding the *RHS* (*right-hand-side*) of the rule in business processes under the condition that these processes also contain the *LHS* (*left-hand-side*).

The Apriori algorithm must provide the *minimum support threshold* (*minSupp*) and the *minimum confidence threshold* (*minConf*) to verify the activityset (as itemset). If the occurrence frequency of the itemset is greater than or equal to *minSupp,* an activityset satisfies the *minSupp*. If an activityset satisfies the *minSupp*, it is a *frequent activityset*. Rules that satisfy both a *minSupp* and a *minConf* are called *strong*.

The algorithm can be explained following [12, 13].

```
Let's define:
    $C_k$ as a candidate activityset of size $k$
        /* activityset is a set of business activities */
    $L_k$ as a frequent activityset of size $k$
```

[4] In order to avoid confusion, we also refer to the reduced versions of process designs with gateways as process designs.

Main steps of iteration are:

1) Find frequent set L_{k-1}
2) Join step:
3) C_k is generated by joining L_{k-1} with itself (Cartesian product L_{k-1} x L_{k-1})
4) Prune step (apriori property):
   Any (*k* – *1*) size *activityset* that is not frequent cannot be a subset of a frequent *k* size *activityset*, so it should be discarded
5) Frequent set L_k has been achieved

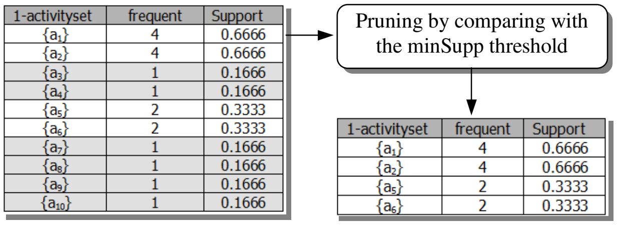
III. An Example of Our Approach

This example uses the collection of process designs in Figure 2 to illustrate how our approach works in practice. After pre-processing these processes, let $P = \{p_1, p_2, p_3, p_4, p_5, p_6\}$ be a set of processes and let $A = \{a_1, a_2,.., a_{10}\}$ be the set of activities. See in Table 1, each row can be taken as a process. We can identify business rules from these models by the support-confidence framework. This example provides the minSupp as 25% and the minConf as 100%.

Using the support-confidence framework, we present a two-step of the Apriori algorithm as follows.

Firstly, the 1-activityset $\{a_1\}$, $\{a_2\}$,…, $\{a_9\}$ and $\{a_{10}\}$ are generated as candidates at the first pass over the collection. The first step of the support-confidence framework is to count the frequencies of *1*-activity. In Table1, the activities $\{a_1\}$ and $\{a_2\}$ occur in four processes, p_1, p_2, p_5 and p_6. Their frequencies are four and their supports are 66%. Meanwhile, processes $\{a_5\}$ and $\{a_6\}$ occur in two processes: p_3 and p_4. Their frequencies are two and their supports are 33%. Consider activities $\{a_3\}$, $\{a_4\}$, $\{a_7\}$, $\{a_8\}$, $\{a_9\}$ and $\{a_{10}\}$ occur in only one process. Thus, their minSupp are 16%.

The second step of the support-confidence framework is to compare the supports of these activitysets with the minSupp. If they are greater than minSupp, they may be strong rules which should be kept for the next step. Meanwhile, if they are less than the minSupp, they are pruned (See in Figure 4).

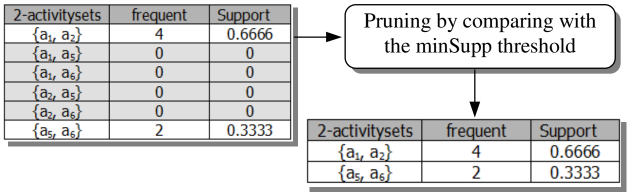


1-activityset	frequent	Support
{a1}	4	0.6666
{a2}	4	0.6666
{a3}	1	0.1666
{a4}	1	0.1666
{a5}	2	0.3333
{a6}	2	0.3333
{a7}	1	0.1666
{a8}	1	0.1666
{a9}	1	0.1666
{a10}	1	0.1666

1-activityset	frequent	Support
{a1}	4	0.6666
{a2}	4	0.6666
{a5}	2	0.3333
{a6}	2	0.3333

Figure 4. The first consideration with *1*-activityset and pruning done by the Apriori algorithm

Secondly, the frequent *1*-activityset $\{a_1\}$, $\{a_2\}$, $\{a_5\}$, and $\{a_6\}$ can be added to a *frontier set* (*F*) for the next pass, and the second pass begins over the collection to search for 2-activites candidates based on the Cartesian product[5]. Each such candidate is subset of *F*, and the 2-activitysets candidates are $\{a_{1,}\ a_2\}$, $\{a_{1,}\ a_5\}$, $\{a_{1,}\ a_6\}$, $\{a_{2,}\ a_5\}$, $\{a_{2,}\ a_6\}$, and $\{a_{5,}\ a_6\}$. In Table 1, the frequent 2-activitysets $\{a_1, a_2\}$ occurs in four processes: p_1, p_2, p_5 and p_6. Its frequency is four and its support is 66%, which is greater than the minSupp. Furthermore, the frequent 2-activitysets $\{a_5, a_6\}$ occurs in two processes: p_3 and p_4. Its frequency is two and its support is 33%, which is greater than the minSupp. However, the frequent *2*-activitysets $\{a_1, a_5\}$, $\{a_1, a_6\}$, $\{a_2, a_5\}$, and $\{a_2, a_6\}$ have not occurred in any process. Thus, they are pruned. Finally, there are only $\{a_{1,}\ a_2\}$ and $\{a_{5,}\ a_6\}$ as frequent activitysets. They are presented in Figure 5.



2-activitysets	frequent	Support
{a1, a2}	4	0.6666
{a1, a5}	0	0
{a1, a6}	0	0
{a2, a5}	0	0
{a2, a6}	0	0
{a5, a6}	2	0.3333

2-activitysets	frequent	Support
{a1, a2}	4	0.6666
{a5, a6}	2	0.3333

Figure 5. The second consideration with 2-activitysets and pruning done by the Apriori algorithm

Thirdly, the frequent *1*-activityset and the frequent *2*-activitysets are appended into the frontier set *F*, and the third pass begins over the dataset to search for *3*-activitysets candidates. There is no frequent 3-activitysets, and the algorithm is ended.

Finally, based on this example, it can be concluded that there are two strong rules which are extracted as business rules: $\{a_{1,}\ a_2\}$ and $\{a_{5,}\ a_6\}$. They imply that $a_{1,}\ a_2$, $a_{5,}$ and a_6 may be "significant" activities in this business. In addition, the association of activities $\{a_{1,}\ a_2\}$ is, "if the activities a_1 is performed, the following activity which should be performed is a_2". Similarly, the association of activities $\{a_{5,}\ a_6\}$ is, "if the activities a_5 is performed, and then the activity a_2 will be performed".

IV. The Experimental Results And Discussion

This section is to present the experimental results with a collection of business process designs from a financial organization.

A. *Collection of Business Process Designs*

The process designs that we used for our experiments capture "*asis*" work-flows. Each process was modeled based on a series of interviews with various actors within a financial organization. Also, each model has been created both in text as well as at various abstraction layers using the

[5] The Cartesian product of two sets *A* and *B* (also called the product set, or cross product) is defined to be the set of all points (*a*, *b*) where $a \in A$ and $b \in B$. It is denoted *A*×*B*, and is called the Cartesian product since it originated in Descartes' formulation of analytic geometry.

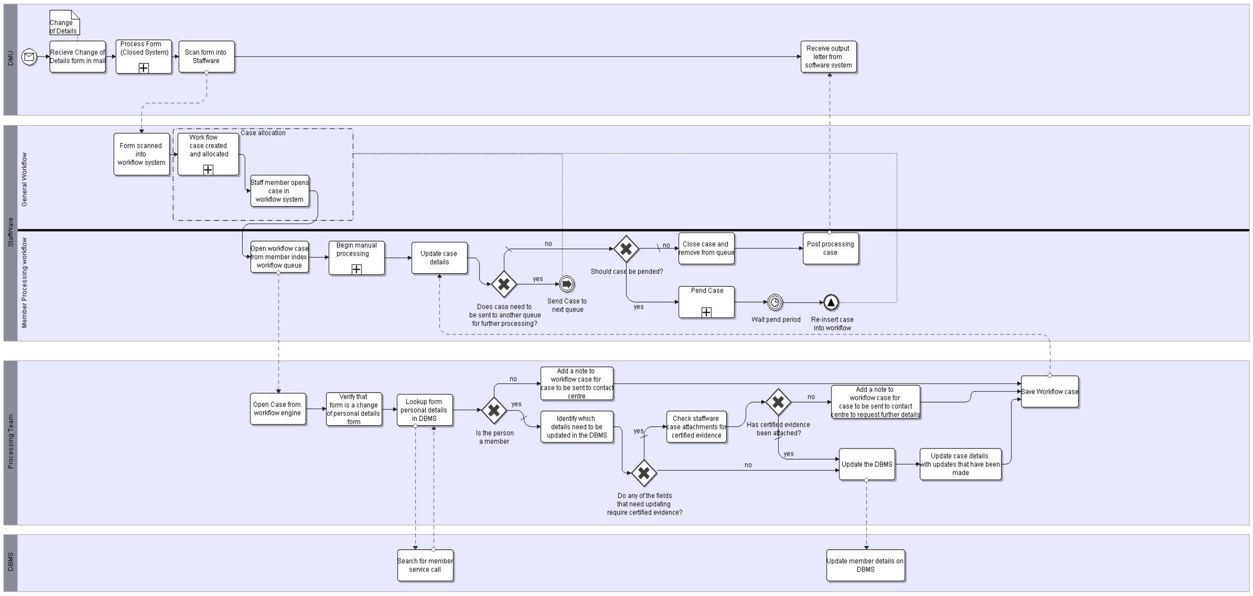

DMU
Change of Details
Recieve Change of Details form in mail
Process Form (Closed System)
Scan form into Staffware
Receive output letter from software system
StaffWare
General Workflow
Form scanned into workflow system
Case allocation
Work flow case created and allocated
Staff member opens case in workflow system
Member Processing workflow
Open workflow case from member index workflow queue
Begin manual processing
Update case details
no
yes
Does case need to be sent to another queue for further processing?
Send Case to next queue
Should case be pended?
no
Close case and remove from queue
Post processing case
yes
Pend Case
Wait pend period
Re-insert case into workflow
Processing Team
Open Case from workflow engine
Verify that form is a change of personal details form
Lookup form personal details in DBMS
Is the person a member
no
Add a note to workflow case for case to be sent to contact centre
yes
Identify which details need to be updated in the DBMS
yes
Check staffware case attachments for certified evidence
Has certified evidence been attached?
no
Add a note to workflow case for case to be sent to contact centre to request further details
yes
Save Workflow case
no
Do any of the fields that need updating require certified evidence?
Update the DBMS
Update case details with updates that have been made
DBMS
Search for member service call
Update member details on DBMS

Figure 6. An example of business process design based on BPMN

Business Process Modeling Notation (BPMN). These models were based on research being undertaken by the Decision Systems Laboratory (DSL), Faculty of informatics, University of Wollongong. An example of business process designs is shown in Figure 6.

B. A Measurement of Business Rules

It is noted that not all strong association rules (also referred to as business rules) are interesting enough to be used [14, 15]. In general, rules can be classified into two types [14]: *negative association rules* and *positive association rules*. A negative association rule also describes relationships between item sets and implies the occurrence of some item sets characterized by the absence of others. In contrast to negative association rules, the classical rules which are most frequently studied in the literature are called positive association rules here.

In our work, we adopt a technique to assess an interesting rule which is called the heuristics technique [15] based on the statistics behind the data to measure association rules. The condition of measurement is presented as follows.

$$A \Rightarrow B \textit{ is interesting rules:}$$
$$\textit{if support } (A, B) - (\textit{support } (A) \times \textit{support } (B)) > 0$$
$$\textit{or} \geq \textit{min-interest}$$

An example of a misleading ‘*strong*’ rules, suppose we are interested in analysing the collection of business process designs in Figure 2 with respect to the association of activites a_1 and a_2. Of the six business processes analysed, the collection shows that four of processes included activity a_1, while four included activity a_2, and four included both a_1 and a_2. Discovering association rules is run on the data collection using a minSupp as 25% and the minConf as 100%. The following association rule is generated as a strong business rule.

$$\textit{activity}(a_1) \Rightarrow \textit{activity}(a_2)\ [\textit{support} = 66\%,\ \textit{confidence} = 100\%]$$

Therefore,

$$\textit{support } (a_1 \Rightarrow a_2) - (\text{support } (a_1) \times \textit{support } (a_2)) \quad > 0$$
$$0.66 - (0.66 \times 0.66) \quad > 0$$

This rule can be a strong rule which is reliable for the use. It is called *positive business rule*.

For our experiment with the collection, we can extract four main business rules having the highest support and confidence as business rules. The interested rules are the positive rules which are enough to use as prior knowledge.

It is noted that, although extracted rules are negative rules, this is ineffective for our work because we only concentrate to extract the association of activities in order to pre-define the identifications of the process design collection. In fact, negative association rules are not for the expensive process to discover them [15]. A number of other applications would benefit from negative association rules. If the correlation coefficient between two items is a negative number (e.g. -0.61), the two items are considered negatively correlated. For an example case, in the market basket analysis [15], this measure sheds a new light on the data analysis on these specific items. The rule is misleading but the correlation brings new information that can help in devising better marketing strategies.

TABLE II. THE MEASUREMENT OF EXTRACTED BUSINESS RULES FROM THE COLLECTION OF BUSINESS DESIGNS USING THE APRIORI ALGORITHM

<table>
<tr><th colspan="3">Four business rules having the highest support and confidence</th></tr>
<tr><th>Business Rules</th><th>Positive rule</th><th>Negative rule</th></tr>
<tr><td>1</td><td>√</td><td>-</td></tr>
<tr><td>2</td><td>√</td><td>-</td></tr>
<tr><td>3</td><td>√</td><td>-</td></tr>
<tr><td>4</td><td>√</td><td>-</td></tr>
<tr><td colspan="3">After ranking business rules having the highest support and confidence, 100% of the selected rules are positive rules.</td></tr>
</table>

C. *How to use these rules?*

After extracting a set of business rules, they can be used as a prior knowledge for further analysing, verifying, and modifying process designs. This can lead to efficiency in various ways such as reducing time of verification and improvement of process designs.

Another approach is to use these business rules to classify the process design collection. Process designs having the same business rules will be grouped in the same class. Slimming down the number of process designs contributes to a better analyzing and verifying of the impact of the process designs.

In our illustration, we use business rules for classifying the business designs collection. Let *P* be a set of process designs $\{p_1, p_2, \ldots, p_n\}$, where each process design is represented by activities $\{a_1, a_2, \ldots, a_n\}$which are non class attributes. Let *c* be a class label *C*. A common rule is defined as

$$p \rightarrow c$$

A business rule can be used to classify a process design. Suppose a business rule is the key identity of the class *c*. If a process design contains the business rule, it should be provided into class *c*.

With our approach, the business design collection used in the experiment can be classified into four groups. Indeed, process designs in each group contain the same business rule. The results can be shown in Table 3.

As can be seen in Table 3, the results of classification from three groups (1, 3, and 4) are satisfactory, but the result of the group 2 is very poor. This is because the process designs which should be in the group 2 also contain the business rule which is the identity of group 1. Therefore, it is hard to analysis. However, with the concept '*first come first served*', the process designs will be firstly analyzed by using the business rule of group 1. As this, these process designs are grouped in to the group 1.

In addition, there is another reason that can lead the accuracy of process designs classification being poor. It is possible that a process design may not contain any business rule. Therefore, it cannot provide some process designs into any group.

If we need the high accuracy of classification, there are two ways to solve this problem. First, the number of business rules should be increased in order to elaborately classify process designs. Second, the business rules should be combined with another technique (e.g. the *k*-means clustering) to increase a performance of process designs classification.

TABLE III. THE RESULTS OF PROCESS DESIGNS CLASSIFICATION BY USING FOUR MAIN BUSINESS RULES

Groups	Precision (%)
1	100
2	0
3	85
4	79
Average of Precision	66

V. CONCLUSION

This paper is to propose an alternative approach to process mining where the objective is to extract process constraints or business rules from existing resources. Our approach focuses on an alternative dataset – process designs as opposed to process instances. We have shown that our approach can offer two benefits. Firstly, if we can identify a set of key activities which are important for driving in processes success as a set of business rules, this can lead to efficiency in various ways such as reducing time of verification and improvement of process designs. Since we rank a priority of activities in a process, when a process faces a problem, process designers may need to check the activities which are in a set of business rules. This is because these activities may lead to the failure of a process and consequently it may become a cause of system failure. However, this can indicate that if a problem of process arises from other activities which are not in the set of business rules, it may effect for the process but it may not hinder the overall performance of process. Therefore, these activities can be allowed to continually perform, and they can be re-checked later.

Secondly, our business rules can be used to classify process designs. Process designs having the same business rule will be grouped in the same class. Slimming down the number of process designs contributes to a better analyzing and verifying of the impact of the process designs. However, the error rate of process designs classification increases since some processes do not reflect the business rules which are the identities of the group or some process designs contain many business rules. Therefore, they cannot be classified into the group because these problems are hard to analyze by only using business rules. For this reason, it may lead to the accuracy of process designs classification being decreased. To deal with these problems, we offered two techniques for modifying the accuracy of process designs classification by business rules. First, the number of business rules should be increased in order to classify process designs. Second, the business rules should be combined with another technique (e.g. the *k*-means clustering) to increase a performance of process designs classification.

REFERENCES

[1] K. Bessai1, B. Claudepierre1, O. Saidani1, and S. Nurcan. Context-aware Business Process Evaluation and Redesign. The 9th Workshop on Business Process Modeling, Development, and Support (BPMDS). 2008.

[2] C. Ren, W. Wang, J. Dong, H. Ding, B. Shao, and Q. Wang. Towards a Flexible Business Process Modeling and Simulation Environment. Proceedings of the 2008 Winter Simulation Conference. 2008.

[3] R. Co, S, Lee, and E.W. Lee. Business Process Management (BPM) Standard: A Survey. Business Process Management Journal. Emerald Group Publishing Limited. 15(5). 2009.

[4] S. Muthu, L. Whitman, and S. H. Cheraghi. Business Process Reengineering: A Consolidated Methodology. Proceedings of The 4th Annual International Conference on Industrial Engineering Theory, Applications and Practice. San Antonio, Texas. 1999.

[5] M.L. Rosa, Marcello, M. Dumas, and H.M. A.T. Hofstede. Modelling Business Process Variability for Design-Time Configuration. In: Cardoso, Jorge and van der Aalst, Wil M.P., (eds.) Handbook of Research on Business Process Modeling. Information Science Reference (IGI Global). (In Press). 2008.

[6] Q. Luo. Advancing Knowledge Discovery and Data Mining. 2008 Workshop on Knowledge Discovery and Data Mining (WKDD). Adelaide, Australia. 2008.

[7] O. Folorunso and A.O. Ogunde. Data mining as a technique for knowledge management in business process redesign. Information Management and Computer Security Journal. Emerald Group Publishing Limited. 13(4), pp 274-280. 2005.

[8] W. M. P. van der Aalst, and A. J. M. M Weijters. Process mining: a research agenda. Computers in Industry. Elsevier Science Publishers. 53(3), pp 231 – 244. 2004.

[9] G. Shmueli, N.R. Patel, P.C. Bruce. Data mining for business intelligence: concepts, techniques, and applications in Microsoft Office Excel with XLMiner. John Wiley & Sons. 2007.

[10] R.G. Ross. Expressing business rules. Proceedings of the 2000 ACM SIGMOD international conference on Management of data. 2000.

[11] R. Agrawal, T. Imielinski, and A. Awami. Database Mining: A performance perspective. IEEE transaction of Knowledge and Data Engineering. pp 914-925. 1993.

[12] J. Han and M. Kamber. Data Mining: Concepts and Techniques (2nd edition). Morgan Kaufmann Publishers. San Francisco, CA. 2006.

[13] C. Zhang and S. Zhang. Association Rules: Models and Algorithms. Springer-Verlag, New York. 2002.

[14] X. Yuan, B.P. Buckles, Z. Yuan, and J. Zhang. Mining negative association rules. The proceedings of seventh International Symposium on Computers and Communications. pp 623- 628. 2002.

[15] A. Maria-Luiza Antonie and Z.R. Osmar. Mining Positive and Negative Association Rules: An Approach for Confined Rules. PKDD2004. 2004.