An ontology-based method for improving the quality of process event logs using database bin logs

Shokoufeh Ghalibafan, Behshid Behkamal* and Mohsen Kahani

Ferdowsi University of Mashhad,
Mashhad, Iran
Email: sh.ghalibafan@mail.um.ac.ir
Email: behkamal@um.ac.ir
Email: kahani@um.ac.ir
*Corresponding author

Mohammad Allahbakhsh

University of Zabol,
Zabol, Iran
Email: allahbakhsh@uoz.ac.ir

Abstract: The main goal of process mining is discovering models from event logs. The usefulness of these discovered models is directly related to the quality of event logs. Researchers proposed various solutions to detect deficiencies and improve the quality of event logs; however, only a few have considered the application of a reliable external source for the improvement of the quality of event data. In this paper, we propose a method to repair the event log using the database bin log. We show that database operations can be employed to overcome the inadequacies of the event logs, including incorrect and missing data. To this end, we, first, extract an ontology from each of the event logs and the bin log. Then, we match the extracted ontologies and remove inadequacies from the event log. The results show the stability of our proposed model and its superiority over related works.

Keywords: data quality; process mining; event log; ontology matching; database bin log.

Reference to this paper should be made as follows: Ghalibafan, S., Behkamal, B., Kahani, M. and Allahbakhsh, M. (2020) 'An ontology-based method for improving the quality of process event logs using database bin logs', *Int. J. Metadata Semantics and Ontologies*, Vol. 14, No. 4, pp.279–289.

Biographical notes: Shokoufeh Ghalibafan received the BSc and MSc in Computer Engineering from the Ferdowsi University of Mashhad in 2014 and 2016, respectively. She worked on a project of increasing data quality for software design from 2013 to 2014 in the Software Quality Lab at the Department of Computer Engineering. Then, she joined Web Technology Laboratory in 2014, and continued her research on process mining and data quality for two years. Her research interests include process mining, semantic web, and data cleansing, and ontology matching.

Behshid Behkamal is an Assistant Professor of Computer Engineering and a leading member of Web Technology Lab (WTlab) at the Ferdowsi University of Mashhad, Iran. She received her PhD in 2014 on the Quality Assessment of Linked Open Data from Ferdowsi University of Mashhad. Her main research interests include data quality, open data, and process mining.

Mohsen Kahani is a Professor of Computer Engineering and the head of Web Technology Lab (WTlab) at Ferdowsi University of Mashhad. He received his BS from the University of Tehran, Iran, and his MS and PhD from University of Wollongong, Australia. His research interests include semantic web, natural language processing, and process mining.

Mohammad Allahbakhsh received his PhD degree in Computer Science and Engineering from The University of New South Wales. He is currently an Assistant Professor at The University of Zabol, Iran. He is also an honorary research fellow at Macquarie University, Australia. His main research interests include quality control and data aggregation in human-enabled applications and participatory sensing systems.

1 Introduction

Process mining aims at discovering, monitoring and improving real-world processes by extracting knowledge from event logs readily available in today's information systems (Bose et al., 2013). Process mining comprises three main tasks: (i) process discovery, (ii) conformance and (iii) enhancement (Van der Aalst et al., 2011). Process discovery, the more common task in process mining, produces a process model based on an event log. The second task, process conformance, compares the existing process model with an event log of the same process. The third task, process enhancement, extends and improves an existing process model based on the information acquired from the actual process, as recorded in event logs. Thus, an event log is essentially the starting point of all process mining tasks. It means that the usefulness of the outcome of all these tasks directly depends on the quality of the even log.

The challenges arising in the analysis of process event logs are rooted in the two sources: (i) process characteristics, and (ii) quality of event logs (Bose et al., 2013). The former deals with fine-grained events, case heterogeneity, voluminous data, and concept drift. For instance, fine-grained event logs and less structured processes result in spaghetti-like process models, which are often hard to comprehend. The second group of the challenges stem from data that are either missing, incorrect, imprecise, or irrelevant. Such low-quality data leads to complicated or unreliable results. This is particularly observed when case attributes are missing. Dealing with these challenges is an ever-growing important research area in process mining.

Several approaches, either semantic or non-semantic ones, are proposed to address these challenges. Ly et al. (2012) provided a semantic method that utilises user-defined constraints to resolve the problem of incorrect/irrelevant data. Another method to address the problem of incorrect data using semantic lifting is proposed in Azzini et al. (2013). Also, a different semantic approach is utilised by Richetti et al. (2014) to deal with the cases of voluminous data and fine-granular events. As non-semantic methods, Rogge-Solti et al. (2013) and De Leoni et al. (2015) respectively focus on missing data and case heterogeneity. To the best of our knowledge, none of the existing approaches in the literature considers application of a reliable external data source to improve the quality of process event logs.

In this paper, we propose to use the bin log of the transactional database, as a relevant source, to tackle incorrect/missing data. We extract ontologies from both the event log and the bin log. Then, we match the extracted ontologies for the process of data cleaning. In addition, our approach relies on novel techniques of ontology extraction and matching.

Thus, the main contributions of this article are summarised as follows:

- We use database bin logs for cleaning event logs.
- We propose a novel technique for extracting an ontology from an event log. More precisely, we enrich the ontology extracted from a process event log using databases.
- We develop an algorithm for the purpose of enhancing the accuracy of ontology matching.

The rest of the paper is organised as follows: the state of the art is reviewed in Section 2. We propose our approach in Section 3. The evaluation results are presented and discussed in Sections 4 and 5, and we conclude the paper in Section 6.

2 Related work

In this section, we primarily focus on semantic process mining, but also touch upon non-semantic methods as relevant to our work.

2.1 Semantic process mining

According to recently published systematic mapping studies, combination of ontology and semantic-based approaches significantly improve the quality of the results achieved in different applications (Garcia et al., 2019). Semantic process mining leverages the data semantics arising from an event log. It has three key components: ontologies, references from the elements of the event log into the concepts in the ontology, and ontology reasoners (De Medeiros et al., 2008; Martinez-Cruz et al., 2012). The main component of semantic process mining is the ontology. Ontologies, which are linked to event logs, can improve the quality of process mining and can result in a better analysis of the process. So, we, first, review the works that extract/enrich ontologies from process event logs. Then, we investigate the methods that focus on taking ontological information into account during process analysis.

A variety of techniques are proposed for ontologies extraction. For instance, Nykänen et al. (2015) used log files to produce two ontologies, the product model and process ontologies. Another type of ontology, TOronto Virtual Enterprise (TOVE), is enriched with concepts related to business models Kim and Werthner (2011). In Caetano et al. (2015), an analytical method is used to show that combining process mining with an organisational ontology aids experts in analysing business processes. Similarly, Detro et al. (2016) presented a model for semantically enriching an event log by a domain ontology. In Pedrinaci and Domingue (2007), an event ontology is defined which includes the events that take place during a process. Vaculin and Sycara (2008) proposed a new definition for event ontology which also comprises a classification involving sub-classes related to events.

The literature on using semantic information for improvement of the quality of event log is very broad. For example, Ly et al. (2012) utilised user-defined constraints to resolve the problem of incorrect/irrelevant data. In Azzini et al. (2013), a new concept of Semantic Lifting (SL) is introduced as an aid in process mining. The procedure is implicitly done using conversion of data in a data warehouse to the event log format, which is appropriate for process monitoring in an information system.

Richetti et al. (2014) discovered declarative process models by exploring event logs and show flexible and unstructured processes. The work presents an approach that uses hierarchical semantic relations to reduce the complexity of process models. There are also frameworks proposed for the pre-processing of the event log for purposes such as aggregating the semantic

information (Deokar and Tao, 2015), and extracting event log information based on the ontology of the related domains (Calvanese et al., 2016).

Also, a semi-automatic method named RDB2Log is proposed to generate quality-informed event logs from relational data (Andrews et al., 2020). It requires as input a relational data set and supports the user in selecting event log attributes from the available data columns.

2.2 Non-semantic methods

The second category involves the studies that use non-semantic methods to improve the quality of event logs. These methods can be divided into two categories: probabilistic and non-probabilistic.

As a probabilistic method, Rogge-Solti et al. (2013) used knowledge from process models to present a method for repairing missing events in an event log by using stochastic Petri nets, alignments and Bayesian network. It first uses path probabilities and then Bayesian network. Sani et al. (2018) introduced a conditional method to detect outlier behaviour. It detects and modifies such behaviours in order to become acceptable inputs for process mining algorithms. In Conforti et al. (2018), an automatic method is presented for reordering the events with incorrect timestamps and estimating correct timestamp for such events. Ayo et al. (2017) presented another method for improving event logs. It uses Fuzzy Genetic Mining based on Bayesian Scoring Function (FGM-BSF).

As non-probabilistic methods, Van der Aalst (2015) and De Murillas et al. (2016) extracted event-related data from databases, but not to directly repair event logs. Dunkl (2013) deals with activity sequencing and activity hierarchy to improve process mining results. A pattern-based method is proposed in Suriadi et al. (2017) for improving the quality of event logs and process models. Nguyen et al. (2019) focused on detecting anomalous values and reconstructing missing values in event logs using auto-encoders to learn a model of the relationships among attribute values.

2.3 Guidelines

In addition to the two categories of methods we reviewed, another category of related works includes studies that present guidelines and instructions for logging, discussions of issues of quality of event logs, etc. For example, Van der Aalst (2015) presented 12 instructions based on the attributes and references of events. Mans et al. (2015) classified the issues concerning quality of events logs. In Alspaugh et al. (2014), some issues related to the event logs are identified and their possible solutions are discussed. Devi and Kalia (2015) recommended a set of basic information items that should be recorded in an event log and provides recommendations for improvement in the quality and quantity of logged user activities, which results in a better information collection. Table 1 provides a comparison amongst the studies regarding the ways they improve process mining.

Table 1 Comparison of studies

Ref.	*Method*		*Automation level*	*Challenge*	*Uses relational database*	*Year*
Ly et al. (2012)	Semantic	Non-probabilistic	Semi-automatic	Event log quality	No	2012
Azzini et al. (2013)	Semantic	Non-probabilistic	Semi-automatic	Event log quality	No	2013
Dunkl (2013)	Non-semantic	Non-Probabilistic	Automatic	Event log quality	No	2013
Rogge-Solti et al. (2013)	Non-semantic	Probabilistic	Automatic	Event log quality	No	2013
Richetti et al. (2014)	Semantic	Non-probabilistic	Automatic	Process-related issues	No	2014
Deokar and Tao (2015)	Semantic	Non-probabilistic	Automatic	Process-related issues	No	2015
Van der Aalst (2015)	Non-semantic	Non-probabilistic	Semi-automatic	Event log quality	YES	2015
Calvanese et al. (2016)	Semantic	Non-probabilistic	Semi-automatic	Process-related issues	YES	2016
De Murillas et al. (2016)	Non-semantic	Non-probabilistic	Semi-automatic	Event log quality	YES	2016
Detro et al. (2016)	Non-semantic	Non-probabilistic	Automatic	Event log quality	No	2016
Suriadi et al. (2017)	Non-semantic	Non-probabilistic	Semi-automatic	Event log quality	No	2017
Ayo et al. (2017)	Non-semantic	Probabilistic	Automatic	Event log quality	No	2017
Sani et al. (2017)	Non-semantic	Probabilistic	Automatic	Event log quality	No	2017
Sani et al. (2018)	Non-semantic	Probabilistic	Automatic	Event log quality	No	2018
Conforti et al. (2018)	Non-semantic	Probabilistic	Automatic	Event log quality	No	2018
Nguyen et al. (2019)	Non-semantic	Non-probabilistic	Automatic	Event log quality	No	2019
Andrews et al. (2020)	Non-semantic	Non-probabilistic	Semi-automatic	Event log quality	YES	2020
Proposed Approach	Semantic	Non-probabilistic	Automatic	Event log quality	YES	2020

As shown in Table 1, despite of the effectiveness of semantic approaches in improving the quality of event logs, it has not received enough attention in the literature. Therefore, it is a line of research that needs more investigations.

3 Proposed approach

In this paper we propose to use the databases, as external sources, to improve the event log quality. In the database, a bin log stores all the operations, including Insert, Delete and Update. We use this log to resolve the inadequacies of the event log including incorrect and missing data. We also use ontologies and ontology matching to correct inadequacies of the event log.

Figure 1 shows the overall architecture of our proposed method. As shown, the method includes six phases: (i) Pre-processing, (ii) Case ID discovery and repairing, (iii) Extraction of ontology from the event log, (iv) Extraction of ontology from the database, (v) Ontology matching and (vi) Cleaning the event log.

3.1 Pre-processing

In this phase, we identify and maintain the necessary information contained in the bin log. The bin log encompasses different database operations (e.g., insert, delete and update) together with the corresponding timestamps. An example of a bin log file is shown in Figure 2.

In order to obtain the required information from the bin log, we pre-process the log by removing unnecessary data, specifically redundancies. The result of the pre-processing phase is a file containing the timestamps, operations (insert, delete or update) and their corresponding values. An example of output of this phase for an 'Insert' operation is as follows:

```
#160205 11:12:17
Insert into PCH
requests(BrandReqID, PeriodID,
BrandReqNo, BrandReqDate,
BrandRequesterID, BrandReqType,
BrandReqUnitId)
values ('120',
'205','0','2016/02/05','28','BRAND','57')
```

Figure 1 Proposed approach

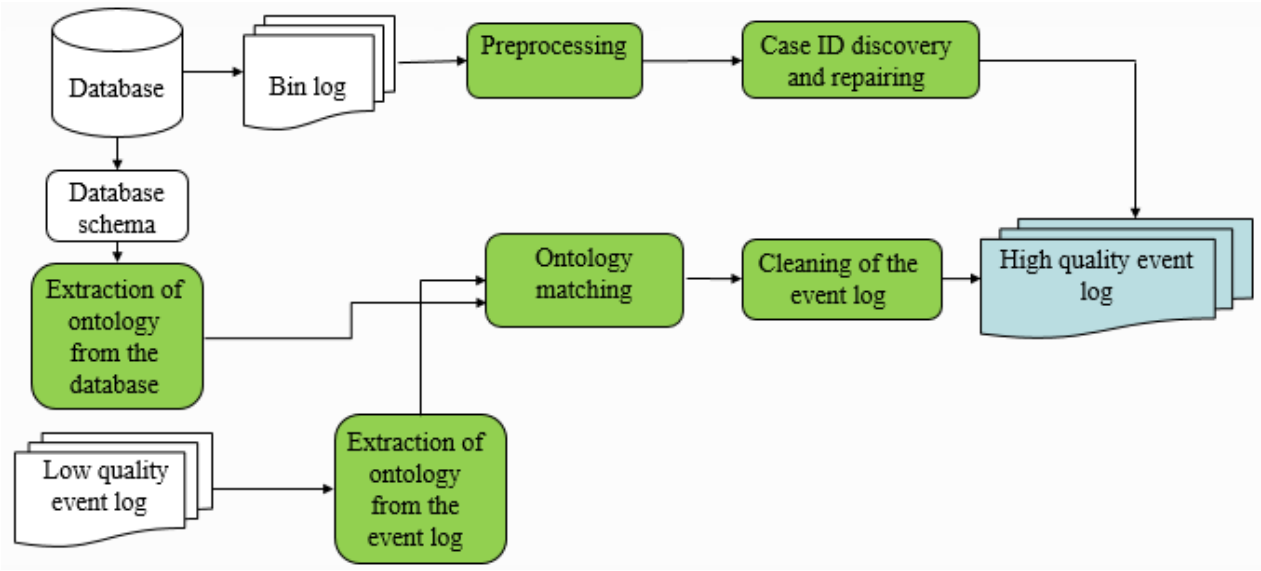


Figure 2 A sample part of the bin log

```
#160205 11:15:47 server id 1  end_log_pos 28759549  Query
thread_id=9313005   exec_time=0 error_code=0
update PCH_requests set BrandReqId='31440',BrandReqDate='2016/02/05'
,BrandReqType='BRAND' where BrandReqId='31440'

#160205 11:15:47 server id 1  end_log_pos 28760290  Query
thread_id=9313005   exec_time=0 error_code=0
insert into PCH_ReqItems(ReqId,GoodID,ItemNumber,ItemPrice,bought,StoreUnitID)
values ('31440','1090391984','1','35000','1','213')
```

Table 2 An example of the number of field repetitions for bin log file

Field \ Table	*BrandReqId*	*BrandReqDate*	*BrandReqType*	*BrandReqId*	*ReqId*	*GoodId*	*ItemNumber*	*ItemPrice*	*bought*	*StoreUnitId*	*RequestId*
T1	1	1	1	1							
T2					4	4	4	4	4	4	
T3											1

3.2 Case ID discovery and repairing

The output of the previous phase is the input of case ID discovery and repairing phase. The aim of this phase is to find the correct case ID. Extracting the process model cannot be done properly with incorrect case IDs in the event log. Repairing the case ID also impacts it's the related fields, such as timestamps.

In order to discover the case ID from a bin log, we compute the number of times that each field has been subject to an operation, in all tables. The result is a table in which rows are table names and columns are all table fields as shown in Table 2. Each cell contains the weighted summation of the times that its corresponding filed has been involved in an operation. The weight of each insert operation is 2 and the weights of both delete and update are considered 1. Then, the highest value which is associated with the primary key is considered for all fields of that table.

3.3 Extracting ontology from the event log

As explained earlier, we extract two ontologies, one from the event log and the other from the database, to match the instances in the event log with those in the database. By instance matching, the incorrect and missing data are repaired using additional information which are provided by ontology. To do so, Java and Jena library are used in order to extract ontologies from the event log and the database. Jena is a Java framework for building semantic web applications, providing extensive Java libraries.[1]

Owing the different structures of the two ontologies, the accuracy of the results obtained at the matching stage may not be desirable if the two ontologies are extracted independently. To overcome this problem, an online mapping should be performed, with the database structure as the reference point, to obtain an accurate matching between the two ontologies.

The first step in extracting ontology from the event log is to define classes corresponding to the fields of the event log and consistent with the database schema. For each field in the event log, a class is created in the ontology. The name given to the class is similar to field's corresponding table. All of the ontology classes are created, in the same way.

The next step is creating the ontology relations. Generally, there are two types of relations in an ontology, i.e., datatype properties and object properties. Datatype properties represent the relations between a class and the value of data, while object properties represent the relations between a pair of classes. In order to create the relations in the ontology, both the relationships between the tables of the database and the relationships between the fields of the event log are considered. By doing so, the ontology of an event log is created through forming classes and establishing relations between them.

After extracting the ontology from the event log, ontology population is carried out. In this step, instances are added to the ontology of the event log in order to match the instances of ontology extracted from the database. For ontology population, the values of event log fields are read one by one, and the instances are inserted into the ontology according to the appropriate class/property.

3.4 Extracting ontology from the database

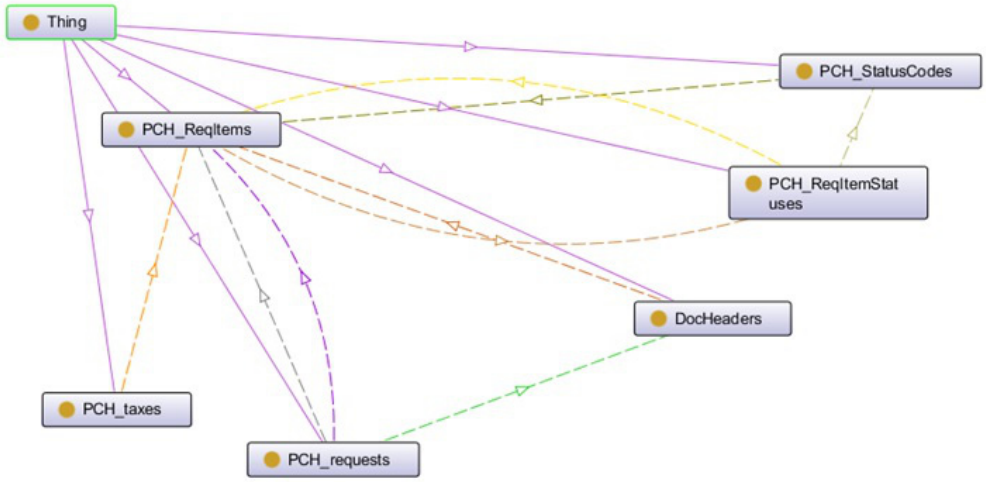
Similar to the extraction of ontology from the event log, another ontology is extracted from the database using the database schema. In this phase, it is necessary to create the classes, object properties and datatype properties. The tables in the database are considered classes, and the fields of each table are considered datatype properties, with their range and domain being determined by considering the database schema. In addition, the relations between the tables (primary and foreign keys) are considered object properties. Ontology population is carried out at this stage, as well. The ontologies extracted from the event log and database are presented in Figures 3(a) and 3(b), respectively.

3.5 Ontology matching

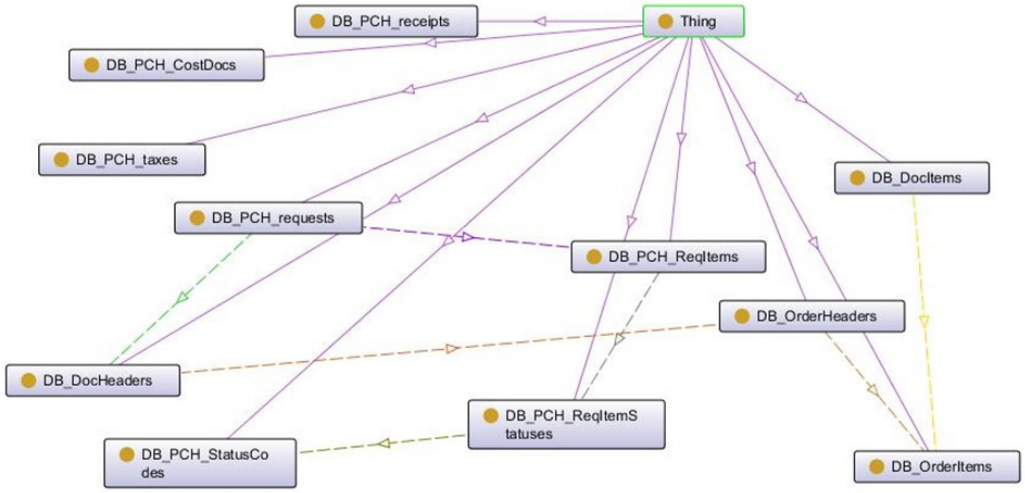
The purpose of ontology matching is to identify the relations between entities from two given ontologies (Euzenat et al., 2015). A variety of methods have been proposed for ontology matching, e.g., matching at the class level, model-based matching and instance matching. In this study, as the goal is to resolve the problem of incorrect and missing data, we perform an exact matching between ontologies.

Missing and incorrect data are related to the datatype properties. So, for accurate matching, equivalent classes and their relations are considered, and exact matching is performed between their corresponding instances. The pseudo-code of our algorithm is shown in Figure 4.

Figure 3 Extracted ontologies. (a) Ontology extracted from the event log (b) Ontology extracted from the database



(a)



(b)

Figure 4 Pseudo-code of the ontology matching algorithm

- **for each** class C_n in EventLog Ontology **do**
  - Extract all instances of class C_n
  - **for each** instance i_n in class C_n **do**
    - Extract all datatype property of i_n
    - Get object property O_n of instance i_n in class C_n and class C_m
    - **for each** datatype property value D_n **do**
      - Get property value v_n of D_n
      - **while**(obtain corresponding class C_p in database ontology with class C_n in EventLog ontology) **do**
        - Get instance i_m in corresponding class C_v in database ontology with class C_m by value of O_n
        - Get object property O_m of instance i_m in class C_v and class C_z
        - Get instance i_p of class C_z with object property O_m
        - **if** class C_z is corresponding class with C_n
          - get property value v_m of D_n in i_p
          - Match(v_n,v_m);
        - **else**
          - **Continue;**

3.6 Cleaning the event log

Generally, there are some mandatory and supplementary items in an event log, i.e., case IDs, events, relationships, case attributes, locations, activity names, timestamps, resources and event attributes. Each of these items may involve data items that are irrelevant, incorrect, imprecise or missing (Bose et al., 2013). Since we are going to clean the event log by comparing it with the database, we can only use the information provided by the bin log of database. So, we exclude the 'resources' item and take into account the remaining eight items.

At this stage, to clean the event log, the following sixteen quality issues are targeted: Missing and incorrect case IDs, Missing and incorrect events, Missing and incorrect relationships, Missing and incorrect case attributes, Missing and incorrect positions, Missing and incorrect activity names, Missing and incorrect timestamps and Missing and incorrect event attributes.

Incorrect and missing data in the ontology of the event log is corrected by matching. In addition, any discrepancy between corresponding datatype properties is handled by substitution of an appropriate value from the database ontology into the event log ontology. This results in clean event log.

4 Empirical evaluation

In this section, we empirically evaluate the performance of our proposed method and show its applicability in the real-world practices. In order to examine the proposed approach and analyse its behaviour, we use a heuristic evaluation method, introduced in Behkamal et al. (2020).

Based on this method, we select a part of an event log and ask domain experts to manually remove quality issues such as missing or incorrect data. As this event log is reviewed and improved by an expert, we use it as a base gold standard for the purpose of performance evaluation. We contaminated this base event log by creating and injecting quality issues, namely, insert incorrect data and create missing data. Then, the proposed method is applied to the manipulated event log. Finally, the result of our approach, which is a repaired event log, is compared with the base expert-reviewed event log and the accuracy of our method is computed. We have asked the director of financial affairs at the Centre of Information and Communication Technology of Ferdowsi University of Mashhad to help us as the domain expert.

In what follows, we first introduce the event log we use for the evaluation purpose. Then, we describe the process of event log contamination, i.e., injecting quality issues to the target event log, and finally present the results.

4.1 The event log

The process which is selected as a case study is the 'Purchase Process' obtained from the Logistics System of Ferdowsi University of Mashhad. We also use the database schema and tables, as well as the event log and the bin log of this system. The specifications of the event log are shown in Table 3.

Table 3 The details of the event log used in this experiment

Number of events	777978
Number of cases	27546
Number of unique events	10910
Number of unique activities	58

It is noteworthy that an event log has a predefined structure, which includes basic information (e.g., case IDs and timestamps). To demonstrate the appropriateness of the selected event log, the size of our selected event log is compared with several standard event logs. The result is shown in Table 4.

Table 4 Comparison of our event log with other event logs

Event log	*Number of cases*	*Number of events*
2011 BPI Challenge (BPIC, 2011)	1143	150291
BPI Challenge (BPIC, 2012)	13087	262200
Catharina Hospital (Bose et al., 2013)	1308	55315
Building Permit Process (Bose et al., 2013)	434	14562
Purchas	777978	27546

4.2 Contaminating the event log

After selecting a part of our experimental even log and cleaning it, we contaminated it using data manipulation method proposed by Nguyen et al. (2019). We randomly insert incorrect data and creation of missing data. To do this, a random function inserts erroneous data into the event log. Initially, 200 records are selected from the event log. Given that each record contains 22 different fields, 4200 data points are considered. These fields are shown in Table 5.

The cells receiving the erroneous data are selected as follows. First, a random integer number in the range [1–200] is generated to be used as the record number and a random number in the range [1–22] as the field number. These two random numbers specify the target cell that will be the target of manipulation. Then, the data existing in the selected cell is eliminated. The data deletion process is repeated 200 times, which produces 200 missing data errors. After this step, incorrect data are inserted into the event log. Once again, the record and field are randomly selected. Considering the selected field (whether it is a number or a string), the data is randomly selected from an invalid array of fields and is placed in the selected field. This is repeated 200 times, resulting in 200 incorrect data issues. Table 5 shows the result of event log contamination.

4.3 Results analysis

In this section, we report the results of applying the proposed method to both base and contaminated event logs.

Experiments show that our proposed method decreases the number of incorrect data to zero, which means full resolution of the problem of incorrect data. However, the problem of missing data is not completely resolved. This is due lack of some information in our database. Therefore, a number of the missing data error remains intact in the event log.

The results shown in Table 6 are obtained for the case of 10% contamination, as described earlier.

Table 5 The number of the inserted erroneous data elements in the fields of the event log

Field	*Incorrect data*	*Missing data*
PeriodYear	8	11
UnitID	3	9
UnitName	11	10
ReqItemId	7	4
Requester	13	14
requesterName	7	4
Actor	10	13
actorName	10	11
StatusCode	7	7
StatusDesc	14	10
ReqReviewDate	13	10
StatusType	7	6
ItemName	3	3
ScaleDesc	9	12
ItemNumber	5	12
IPrice	10	11
TaxDesc	9	8
IDesc	12	11
SupplierId	11	5
Saller	11	9
PurchaserId	12	14
PurchaserName	8	6

Table 6 Final results of empirical evaluation

Metric	*% Missing data correlation*	*% Incorrect data correlation*
Precision	99.88	99.52
Recall	99.95	99.52
*F*1	99.91	99.52

5 Comparative evaluation

In order to demonstrate the superiority of the proposed approach over the other semantic cleaning methods, we have selected a similar work to implement and compare the results. As shown in Table 1, only two of the semantic cleaning methods have focused on the quality improvement of event log (Ly et al., 2012; Azzini et al., 2013), while the others investigated the process-related issues. Azzini et al. (2013) aimed at extracting knowledge about the structure of the process using semantic technologies, so their method cannot be applied to the clean the event log. Ly et al. (2012) presented a data cleaning method that utilises semantic knowledge about the processes in the form of process constraints. The constraints which are imposed by the experts have been directly applied to the event log. Since the method is applicable to our case study, we have selected this as a related work for performance comparison.

To this end, we first need to define the process restrictions by the help of experts. In our work, the constraints are determined by the owner of 'Purchase Process' who is the head of Logistics Systems at Ferdowsi University of Mashhad. A number of process restrictions are defined as follows:

- Preliminary review of the purchase request requires registration, so the registration time must be before the initial review time;
- End state of each process instance must have one of the following values: Suspended, Deleted or Review Appeal;
- If the status of an activity in a process instance is "Reject", for the same process instance, an activity with a "Send" status must have occurred.

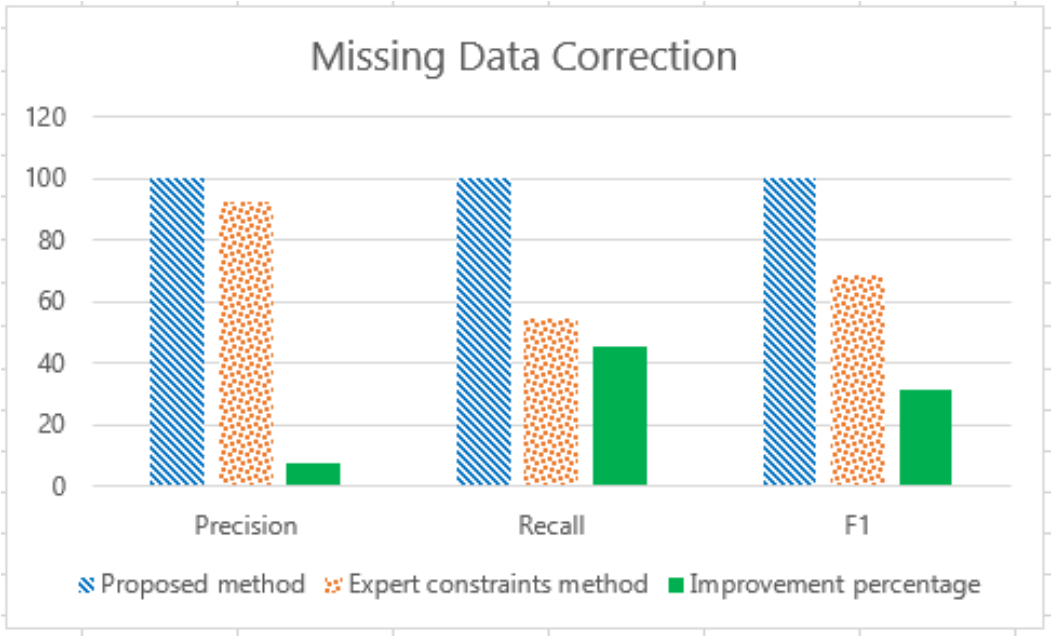
After matching the event log to the above domain-specific constraints, the cases that violate the constraints are removed from the event log. The results are shown by three metrics in Table 7. As shown in this table, the method fails to achieve acceptable precision, recall and *F*-measure in terms of handling missing data. The reason is that some missing data are unintentionally removed as they violate the constraints.

Table 7 Results of evaluation of the method presented by Ly et al. (2012)

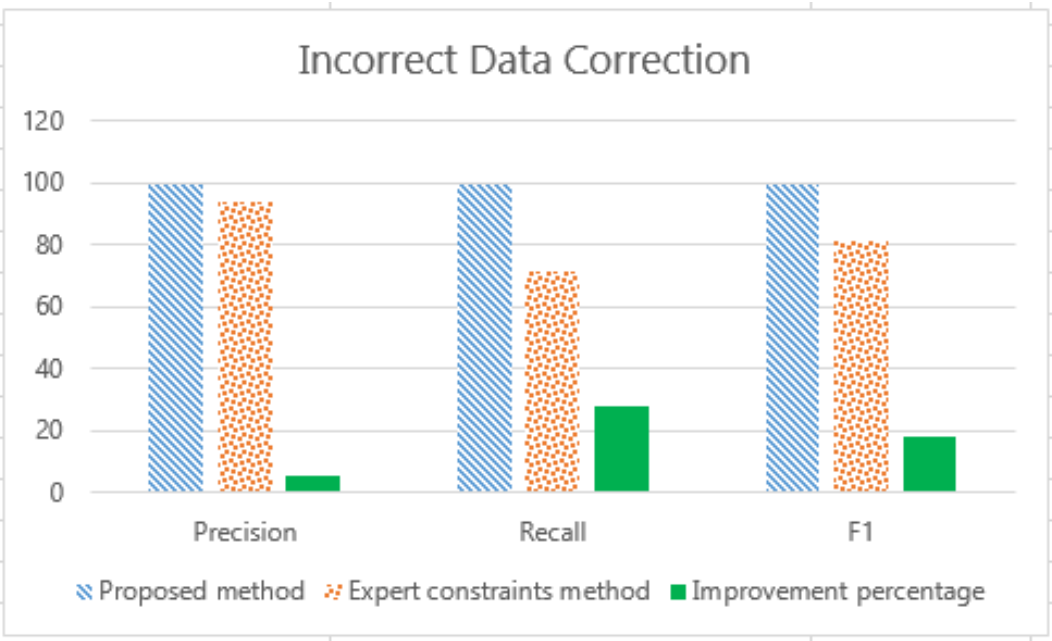
Metric	*% Missing data correlation*	*% Incorrect data correlation*
Precision	92.18	94.04
Recall	54.76	71.42
*F*1	68.70	81.18

The comparison between our proposed method and the expert constraints method is shown in Figure 5(a) for missing data correction and is shown in Figure 5(b) for incorrect data correction.

Figure 5 Comparison between the two methods. (a) Comparison for missing data correction (b) Comparison for incorrect data correction



(a)



(b)

6 Conclusions

In this paper, a novel method is proposed for improving the quality of event logs. Our proposed method employs the bin log of the database, and ontology extraction and matching techniques to improve the quality of an event log. We have validated our method through standard metrics, and discussed its suitability for solving the problem. We have also compared its performance with a related work and shown its superiority over the selected related approach.

The approach proposed in this paper can be applied to resolve the problems of event log in any other domain subject to availability of required input data.

We are going to extend our work in three directions: (i) resolving problem of imprecise data and irrelevant data in the event log; (ii) extending the method to be applicable to large amounts of data, including large event logs and bin logs and (iii) extending the proposed model to be able to clean event log using the bin log of an unstructured database system.

References

Alspaugh, S., Ganapathi, A., Hearst, M.A. and Katz, R. (2014) 'Better logging to improve interactive data analysis tools', *Proceedings of the ACM SIGKDD Workshop on Interactive Data Exploration and Analytics (IDEA'14)*, pp.19–25.

Andrews, R., Van Dun, C.G.J., Wynn, M.T., Kratsch, W., Röglinger, M.K.E. and Ter Hofstede, A.H.M. (2020) 'Quality-informed semi-automated event log generation for process mining', *Decision Support Systems*. Doi: 10.1016/j.dss.2020.113265.

Ayo, F.E., Folorunso, O. and Ibharalu, F.T. (2017) 'A probabilistic approach to event log completeness', *Expert Systems with Applications*, Vol. 80, pp.263–272.

Azzini, A., Braghin, C., Damiani, E. and Zavatarelli, F. (2013) 'Using semantic lifting for improving process mining: a data loss prevention system case study', *Proceedings of the ACM International Symposium on Data-Driven Process Discovery and Analysis (SIMPDA'13)*, pp.62–73.

Behkamal, B., Kahani, M., Bagheri, E. and Sazvar, M. (2020) 'A metric suite for systematic quality assessment of linked open data', *rXiv preprint arXiv:2002.10688*, 2020.

Bose, R.P.J.C., Mans, R.S. and van der Aalst, W.M.P. (2013) 'Wanna improve process mining results?', *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining (CIDM'13)*, IEEE, pp.127–134.

Business Processing Intelligence Challenge (BPIC) (2011) *Business Processing Intelligence Challenge*. Available online at: https://www.win.tue.nl/bpi/doku.php?id=2011:challenge (accessed on 5 May 2020).

Business Processing Intelligence Challenge (BPIC) (2012) *Business Processing Intelligence Challenge*. Available online at: https://www.win.tue.nl/bpi/doku.php?id=2012:challenge (accessed on 5 May 2020).

Caetano, A., Pinto, P., Mendes, C., Da Silva, M.M. and Borbinha, J. (2015) 'Analysis of business processes with enterprise ontology and process mining', *Enterprise Engineering Working Conference*, Springer, pp.82–95.

Calvanese, D., Montali, M., Syamsiyah, A. and van der Aalst, W.M.P. (2016) 'Ontology-driven extraction of event logs from relational databases', *Proceedings of the ACM International Conference on Business Process Management*, Springer, pp.140–153..

Conforti, R., La Rosa, M. and Ter Hofstede, A. (2018) 'Timestamp repair for business process event logs. 2018.

De Leoni, M., Maggi, F.M. and Van der Aalst, W.M.P. (2015) 'An alignment-based framework to check the conformance of declarative process models and to preprocess event-log data', *Information Systems*, Vol. 47, pp.258–277.

De Medeiros, A.K.A., Van der Aalst, W. and Pedrinaci, C. (2008) 'Semantic process mining tools: Core building blocks', *Proceedings of the European Conference on Information Systems*, pp.1–13.

De Murillas, E.G.L., Van der Aalst, W.M.P. and Reijers, H.A. (2016) 'Process mining on databases: unearthing historical data from redo logs', *Proceedings of the ACM International Conference on Business Process Management*, Springer, pp.367–385.

Deokar, A.V. and Tao, J. (2015) 'Semantics-based event log aggregation for process mining and analytics', *Information Systems Frontiers*, Vol. 17, No. 6, pp.1209–1226.

Detro, S.P., Morozov, D., Lezoche, M., Panetto, H., Santos, E.P. and Zdravkovic, M. (2016) 'Enhancing semantic interoperability in healthcare using semantic process mining', *Proceedings of the 6th International Conference on Information Society and Techology (ICIST'16)*, Kopaonik, Serbia, pp.80–85.

Devi, S. and Kalia, A. (2015) 'Study of data cleaning & comparison of data cleaning tools', *International Journal of Computer Science and Mobile Computing*, Vol. 4, No. 3, pp.360–370.

Dunkl, R. (2013) 'Data improvement to enable process mining on integrated non-log data sources', *International Conference on Computer Aided Systems Theory*, Springer, pp.491–498.

Euzenat, J., David, J., Locoro, A. and Inants, A. (2015) 'Context-based ontology matching and data interlinking. 2015.

Garcia, C.D.S., Meincheim, A., Faria, E.R., Dallagassa, M.R., Sato, D.M.V., Carvalho, D.R., Santos, E.A.P. and Scalabrin, E.E. (2019) 'Process mining techniques and applications – a systematic mapping study', *Expert Systems with Applications*, Vol. 133, pp.260–295.

Kim T.T.T. and Werthner, H. (2011) 'An ontology-based framework for enriching event-log data', *The Proceedings of the 5th International Conference on Advances in Semantic Processing*, pp.110–115.

Ly, L.T., Indiono, C., Mangler, J. and Rinderle-Ma, S. (2012) 'Data transformation and semantic log purging for process mining', *Proceedings of the ACM International Conference on Advanced Information Systems Engineering*, Springer, pp.238–253.

Mans, R.S., Van der Aalst, W.M.P. and Vanwersch, R.J.B. (2015) 'Data quality issues', *Process Mining in Healthcare*, Springer, pp.79–88.

Martinez-Cruz, C., Blanco, I.J. and Vila, M.A. (2012) 'Ontologies versus relational databases: are they so different? A comparison', *Artificial Intelligence Review*, Vol. 38, No. 4, pp.271–290.

Nguyen, H.T.C., Lee, S., Kim, J., Ko, J. and Comuzzi, M. (2019) 'Autoencoders for improving quality of process event logs', *Expert Systems with Applications*, Vol. 131, pp.132–147.

Nykänen, O., Rivero-Rodriguez, A., Pileggi, P., Ranta, P.A., Kailanto, M. and Koro, J. (2015) 'Associating event logs with ontologies for semantic process mining and analysis', *Proceedings of the 19th International Academic Mindtrek Conference*, pp.138–143.

Pedrinaci, C. and Domingue, J. (2007) 'Towards an ontology for process monitoring and mining', *CEUR Workshop Proceedings*, Vol. 251, pp.76–87.

Richetti, P.H.P., Baião, F.A. and Santoro, F.M. (2014) 'Declarative process mining: reducing discovered models complexity by pre-processing event logs', *Proceedings of the ACM International Conference on Business Process Management*, Springer, pp.400–407.

Rogge-Solti, A., Mans, R.S., Van der Aalst, W.M.P. and Weske, M. (2013) 'Repairing event logs using timed process models', *Proceedings of the ACM OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*, Springer, pp.705–708.

Sani, M.F., Van Zelst, S.J. and Van der Aalst, W.M.P. (2017) 'Improving process discovery results by filtering outliers using conditional behavioural probabilities', *Proceedings of the ACM International Conference on Business Process Management*, Springer, pp.216–229.

Sani, M.F., Van Zelst, S.J. and Van der Aalst, W.M.P. (2018) 'Repairing outlier behaviour in event logs', *Proceedings of the ACM International Conference on Business Information Systems*, Springer, pp.115–131.

Suriadi, S., Andrews, R., Ter Hofstede, A.H.M. and Wynn, M.T. (2017) 'Event log imperfection patterns for process mining: towards a systematic approach to cleaning event logs', *Information Systems*, Vol. 64, pp.132–150.

Vaculin, R. and Sycara, K. (2008) 'Semantic web services monitoring: an owl-s based approach', *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, IEEE, pp.313–313.

Van der Aalst, W.M.P. (2015) 'Extracting event data from databases to unleash process mining', *BPM-Driving Innovation in a Digital World*, Springer, pp.105–128.

Van der Aalst, W.M.P., Adriansyah, A., De Medeiros, A.K.A., Arcieri, F., Baier, T., Blickle, T., Bose, J.C., Van Den Brand, P., Brandtjen, R. and Buijs, J. et al. (2011) 'Process mining manifesto', *Proceedings of the ACM International Conference on Business Process Management*, Springer, pp.169–194.

Note

1 https://jena.apache.org/