

12-1-2017

PROCESS MODELS DISCOVERY AND TRACES CLASSIFICATION: A FUZZY-BPMN MINING APPROACH.

Kingsley Okoye Dr

University of East London, kingsley.okoye@tec.mx

Usman Naeem Dr

University of East London, UK, u.naeem@uel.ac.uk

Syed Islam Dr

University of East London, UK, syed.islam@uel.ac.uk

Abdel-Rahman H. Tawil Dr

University of East London, UK, A.R.Tawil@uel.ac.uk

Elyes Lamine Dr

Université de Toulouse, Mines-Albi, France, Elyes.Lamine@mines-albi.fr

Follow this and additional works at: <https://scholarworks.lib.csusb.edu/jitim>



Part of the [Artificial Intelligence and Robotics Commons](#), [Business Intelligence Commons](#), [Computer and Systems Architecture Commons](#), [Data Storage Systems Commons](#), [Management Information Systems Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Software Engineering Commons](#), [Technology and Innovation Commons](#), and the [Theory and Algorithms Commons](#)

Recommended Citation

Okoye, Kingsley Dr; Naeem, Usman Dr; Islam, Syed Dr; Tawil, Abdel-Rahman H. Dr; and Lamine, Elyes Dr (2017) "PROCESS MODELS DISCOVERY AND TRACES CLASSIFICATION: A FUZZY-BPMN MINING APPROACH.," *Journal of International Technology and Information Management*: Vol. 26: Iss. 4, Article 1. DOI: <https://doi.org/10.58729/1941-6679.1337>
Available at: <https://scholarworks.lib.csusb.edu/jitim/vol26/iss4/1>

This Article is brought to you for free and open access by CSUSB ScholarWorks. It has been accepted for inclusion in *Journal of International Technology and Information Management* by an authorized editor of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.

Process Models Discovery and Traces Classification: A Fuzzy-BPMN Mining Approach.

Kingsley Okoye

Usman Naeem

Syed Islam

Abdel-Rahman H. Tawil

School of Architecture Computing & Engineering, University of East London,
Docklands Campus, E16 2RD

{K.Okoye, U.Naeem, S.Islam, A.R.Tawil}@uel.ac.uk

UNITED KINGDOM

Elyes Lamine

Université de Toulouse, Mines-Albi, CGI, Campus Jarlard, Albi Cedex 09, France

Elyes.Lamine@mines-albi.fr

FRANCE

ABSTRACT

The discovery of useful or worthwhile process models must be performed with due regards to the transformation that needs to be achieved. The blend of the data representations (i.e data mining) and process modelling methods, often allied to the field of Process Mining (PM), has proven to be effective in the process analysis of the event logs readily available in many organisations information systems. Moreover, the Process Discovery has been lately seen as the most important and most visible intellectual challenge related to the process mining. The method involves automatic construction of process models from event logs about any domain process, and describes causal dependencies between the various activities as performed within the process execution environment. In principle, one can use process discovery to obtain process models that describes reality. To this end, the work in this article presents a Fuzzy-BPMN mining approach that uses training events log representing 10 different real-time business process executions to provide a method for discovery of useful process models, and then cross-validating the derived models with a set of test event logs in order to measure the accuracy and performance of the employed approach. The method focuses on carrying out a classification task to determine the traces, i.e. individual cases that makes up the test event logs in order to determine which traces that can be replayed by the original model. Thus, the paper aim is to provide a technique for process models discovery which is as good in balancing between “overfitting” and “underfitting”

as it is able to correctly classify the traces that can be replayed (i.e allowed) or non-replayable (disallowed) by the model. In other words, the study shows through the Fuzzy-BPMN replaying notation and the series of validation experiments - how given any classified trace (for the test events log) and discovered process model (the training log) it can be unambiguously determined whether or not the traces found can be replayed on the discovered model.

KEYWORDS: process mining, process discovery, classifiers, fuzzy models, BPMN notation, event logs, classification, process models

INTRODUCTION

The need for novel approaches in design and integration of computational intelligence and technologies into everyday (e.g. business) processes, have sprout new insights and unceasing research investigations particularly on how to exploit such tools for use in improving the various organisational processes (Van der Aalst, 2016; Van der Aalst et al, 2010). In recent years, a common challenge with many of the business processes has been on how to develop intelligent systems and/or techniques that can provide platforms for exploring the additional, and most often, the monotonous tasks of managing the entire operational process and quality of information - by providing understandable and useful insights on the best possible ways to make the envisioned information explicable in reality using the underlying events log recorded in the IT systems.

Most organizations have invested in projects to model their various operational processes. However, majority of the derived process models are often unfitting, non-operational, or represents a form of reality that are pointed towards comprehensibility rather than covering the entire actual business process complexities. Therefore, the ability to mine useful or worthwhile knowledge from the readily extracted datasets in current information systems appears to be a challenge, due to the exponential increase in the volume of data that is generated. In consequence, this has spanned the need for a richer and advanced description of real-time processes that allows for flexible exploration of the large volumes of data targeted at improving the system performance and analysis.

Even more, researchers (Dou, et al., 2015; de Medeiros & Van der Aalst, 2009; Van der Aalst, 2016) have shown that a better way of attaining a closer look at any organisation's operational process is to consider the events log that are readily available in its process-base or IT systems. Perhaps, an accurate exploration and/or analysis of the events log could provide vital and valuable information with regards to the quality of support being offered for the so-called organizations and their information systems at large. For example, revealing the underlying relationships

the process elements or individual actors share amongst themselves within the knowledge-base. Such process-related analysis, often allied to the *process mining*, means there is also need for tools and techniques that are capable of extracting valuable information from the event logs about the real-time processes. Practically, there are two main drivers for such growing interest in the process mining field. On the one hand, more and more events are being recorded, thus, providing detailed information about history of processes as they happen in reality. On the other hand, there is need to improve and support business processes in a competitive and rapidly changing environment (Van der Aalst, 2016). Thus, process mining (PM) means extracting valuable, process-related information from event logs about any real time process.

Recently, the Process Mining (Van der Aalst, 2016) has become a valuable technique used to discover such meaningful information from the event data logs. Besides, the PM field combines techniques from computational intelligence which has been lately considered to encompass artificial intelligence (AI) or even the latter, augmented intelligence (AIs) systems, and data mining (DM) to *process modelling*, as well as several other disciplines to analyze the events logs. Indeed, since the PM techniques builds on computational intelligence and data mining techniques, which has led to its significant influence on how process owners and analysts perceive and analyse the readily available large volumes of data captured from their various IT systems. Besides, a greater number of the resulting models and methods tends to support not just *machine-readable* systems but also *machine-understandable* systems. By machine-understandable systems we refer to methods that are developed not just for representing information in formats that can be easily understood by humans, but also for creating applications and/or systems that trails to inclusively process the information that they contain or supports.

Furthermore, the *Classification* - according to (Han and Kamber, 2005) is one of the most universally data mining technique that aims at finding models or functions that describes or distinguishes data attributes or concepts. Specifically, the authors in (Elhebir and Abraham, 2015) notes that pattern discovery algorithms makes use of statistical and machine-learning techniques to build models that predicts behaviour of captured datasets, and concedes that one of the most pattern discovery techniques used to extract knowledge from pre-processed data is *classification*. The authors observe that most of the existing classification algorithms attains good performance for specific problems but are not robust enough for all kinds of discovery problems, and further propose that combination of multiple classifiers (i.e. Hybrid Intelligent Systems (HIS) such as the Fuzzy-BPMN miner proposed in this paper) could be considered as a general solution for the pattern discovery

because they obtain better results compared to a single classifier as long as the components are independent and/or have diverse outputs.

In turn, this paper trails to make use of such valuable, process-related analysis and capabilities of the PM technique and the classification method to analyse data about a real time business process provided by the IEEE CIS Task Force on Process Mining (Carmona, et al., 2016) in order to show the usefulness and impact of the proposed approach in this paper, namely: the Fuzzy-BPMN miner. In other words, this paper looks at the practical use of such techniques related to the process mining to propose a method that is used to extract meaningful patterns from the event logs captured about those processes, and ways of transforming and analysing the datasets into effective minable formats in order to provide meaningful and worthwhile understanding of the processes as performed in reality.

To this end, the work outlines in the following sub-section - the research context and scope of study including the problems which the paper pursues to address and how it is related in context of the research experimentations and proposals.

Research questions

Primarily, the work in this article explores the best possible ways towards the:

***RQ1:** Use of process mining techniques to discover, monitor and analyse event logs about any domain process by discovering useful and worthwhile process models?*

***RQ2:** By what method to determine the extent to which the classification process is able to accurately classify the individual traces that can be found within the event logs and the derived process models?*

Fundamentally, to address the *RQ1* and *RQ2*, the work utilizes the data about a real-time business process provided by the IEEE CIS Task Force on Process Mining (Carmona, et al., 2016) to show how one can efficiently mine and analyse the sets of unobserved behaviours or patterns (i.e the process instances) that can be found within the event logs in order to discover useful and worthwhile process models. Also, the paper discusses the replaying semantics of the process modelling notations that has been employed, and then provide a description of the tools used to discover the process models as well as evaluation of the results of the classification task. Above all, the work looks at the sophistication of the proposed Fuzzy-BPMN mining approach, validation of the classification tasks, and the discovered process models.

Research aim

The overall goal of the work carried out in this paper is to:

“extract streams of event logs from any given domain process (case study of the Business Process data from the IEEE CIS Task Force on Process Mining) and describe formats that allows for mining and improved process analysis of the captured data”.

In other words, the focus of this article is to:

- apply process mining techniques to a given domain process e.g. the Business Process, and
- to provide minable formats and understanding about the available datasets (i.e event logs) as well as useful strategies towards the development of process mining techniques/algorithms that exhibits a high level of accuracy for the classification of the individual traces that can be found within the events log and the discovered models.

Research objectives

Practically, this work uses the case study of the real-time Business Process provided by the IEEE CIS Task Force on Process Mining (Carmona et al, 2016) to seek ways on *how* to do the following:

- RO1** *Extract data from process domains to show how we harmonize and provide events log formats for any given process domain.*
- RO2** *Transform the extracted data into minable executable formats to support the discovery of valuable process models through the proposed technique in this paper.*
- RO3** *Provide techniques for accurate classification of unseen process instances (traces) that can be found within the events log and the derived process models.*
- RO4** *Assess and evaluate the level of accuracy of the classification process by the proposed method in this paper through further analysis of the discovered models.*
- RO5** *Importance of the process mining technique to interpret/support process-related analysis and enhance information value of data about any domain process: case study of the real-time business process data from the IEEE CIS Task Force on Process Mining.*

In principle, this article explores the technological potentials and prospects of using the proposed Fuzzy-BPMN mining approach to addresses a typical process

discovery problem in (Carmona, et al., 2016) (as explained in details in the use case scenario and problem statement section of this paper) - by providing a method that combines the capability of the *Fuzzy mining* algorithms which directly addresses the problem of large numbers of activities and highly unstructured data to show understandable models for very unstructured processes (thus produces simplified process models) and the *Business Process Modeling Notation* (BPMN) which have proven to be useful towards construction of process models with notational elements that are capable of describing the nesting of individual activities (i.e process instances) by using the event-based split and join gateways (i.e. AND, XOR, and OR etc). Thus, the proposal of the Fuzzy-BPMN miner. In other words, the work introduces by means of the proposed Fuzzy-BPMN miner - a process discovery approach that proves useful towards discovering of new and meaningful process models based on the captured events logs (using the case study of the data about a business process provided by the IEEE CIS Task Force on Process Mining) without any prior information on how those activities are performed. Indeed, the outcome of the research experimentations and data validation (as described in the subsequent experimental section of this paper) shows that the proposed process mining approach has correctly classified to a high percentage the accuracy of the individual traces that can be found within the original process models. Thus, determines the traces which can be replayed (i.e allowed or fitting the model), and the traces which are non-replayable (disallowed or not fitting the model).

Accordingly, this article presents the rest of the paper and its results in the following order:

- 1: Background Information.
- 2: The Use Case Scenario and Problem Statement.
- 3: Fuzzy-BPMN Mining Approach: Method, Algorithms & the Classification task.
- 4: Classified Traces Replay and Model Fitness Calculation
- 5: Results and Outcome of the Fuzzy-BPMN Mining Approach
- 6: Discussions and Limitations
- 7: Conclusion

BACKGROUND INFORMATION

Process mining (PM) research started at the Eindhoven University of Technology (TU/e) in 1999, and was first proposed by Wil van der Aalst (Van der Aalst, et al., 2003; Van der Aalst, et al., 2004). According to (Van der Aalst, 2016) as of then, there were limited availability of event logs, and the early methods used to perform

process mining tasks at that time were exceptionally ineffective and naive. Interestingly, for the past few decades, the process mining tools and approaches has undisputedly matured because event data logs has become ever more available, thanks to the Big Data initiative (Van der Aalst, 2016). Moreover, progress has been spectacular within the process mining field and the technique is currently being supported by various tools and algorithms such as the one introduced in this study. In recent time, the author in (Van der Aalst, 2016) describes the process mining term as one of the main mechanisms of “Data Science”. The author opines that process mining has the capacity to provide methods towards bridging the gap between *data science* and *process science*. According to the author, Process Science has emerged due to the process-perspective that is missing in most big data initiative and the wider curricula of data science. Besides, the author in (Van der Aalst, 2011; Van der Aalst, 2016) argue that the events data logs extracted and stored in many organisations IT system must be utilised to enhance the end-to-end processes in reality by focusing on analysing the unseen behaviours based on the information that are present in the logs, thus, the emergence of *process mining*.

Furthermore, whilst the initial attention was primarily on the *process discovery*, the PM field have significantly widened, for instance, the conformance checking, operational support, and multi-perspective process mining which has now grown into fundamental part of many tools and approaches that supports the extraction, modelling and/or interpretation of processes. Particularly, ProM (Verbeek, et al., 2011) one of the leading process mining tool currently in literature.

Nowadays, several organisations have focused on applying the process mining to different aspects of their business processes management and operations. Moreover, the application of the PM techniques are not only or limited to business processes, but also provides new means to discover, monitor, and enhance any given process domain or interest (De Leoni & Van der Aalst, 2013; Van der Aalst, et al., 2012). According to (Van der Aalst, 2011) there are two main drivers for the growing interest in process mining. First, data about many organizations business processes are captured and stored at an unprecedented rate. Secondly, there is ever increasing need to improve and support business processes in a competitive and rapidly changing environments. This means that - process mining have likewise proved its relatability and application in some other field areas including: Health care (Rojas, et al., 2016), Government sectors (Van der Aalst, 2016), Banking and Financial industries (Jans, 2011; Van der Aalst, et al., 2010), Educational organizations and settings (Cairns, et al., 2014; Okoye, et al., 2016), Airlines and Transportation industry (Van der Aalst, 2016), ICT and Cloud Computing (Chesani, et al., 2016) etc. Indeed, the PM techniques uses event data from any these process domains to discover process models, perform conformance checking

of the discovered models, analyse deviations, and even more, extend and predict future outcomes and/or developments.

Actually, many explanations of the PM notion has been propose in literature. Reference (Van der Aalst, 2011) refers to the process mining - as a young research field that makes use of the data mining (DM) technique to find out patterns or models from event logs, and predict outcomes through further analysis of the discovered models. According to the author (Van der Aalst, 2011; Van der Aalst, 2016) PM means extracting valuable, process-related information from event logs about any domain process.

Reference (Cairns, et al., 2015) also mentions that the PM term is concerned with analysis of captured events log from a process-perspective. Reference (Ingvaldsen, 2011) states that as soon as a particular process (e.g. business process) is being supported by some form of IT system, its operational transactions or activities executions can then be observed or recorded in the form of event logs. Likewise, references (Greco, et al., 2006; Van der Aalst, 2011) notes that the process mining notion is an attempt towards extraction of meaningful and non-trivial information from recorded event logs.

Notably, the lion's share of attention in process mining has been devoted to the *process discovery* - i.e., extracting process models, mainly business process models from recorded events log (Carmona, et al., 2016). The Process discovery has been lately seen as the main significant and furthestmost challenge logically allied to the PM term (Carmona, et al., 2016; Van der Aalst, 2011). Process discovery techniques aims to automatically construct process models, e.g., BPMN, Petri-nets, C-nets, Fuzzy models, Process Trees etc. (Van der Aalst, 2016) from events log about a process, and describes causal dependencies between the individual activities as performed in reality. In short, a typical process discovery method takes (as input) recorded event logs, and then produce (as output) a model without any prior information on how the activities has been formerly performed. Besides, in settings where the datasets (i.e. event logs) includes information about resource (e.g. roles), it is also possible to discover resource-related models. For instance, a shared neural network model representing how employees works collectively or collaborate within a particular organisation. In essence, one can make use of the *process discovery* methods to obtain models that describes reality.

More so, the *conformance checking* is the second type of process mining techniques. The method focuses on determining (assessing) how fit the discovered process models describes the actual observation in the event logs (Ingvaldsen, 2011) such as the approach described in this paper. Principally, a conformance

check and analysis technique references an a-priori (i.e. existing) process model and compares it with the events log of the specific (i.e. the same) process. Clearly, such analysis is performed in order to check if in reality, the recorded events data log conforms to the deployed process models (Munoz-Gama & Carmona, 2011; Adriansyah, et al., 2011; Rozinat & Van der Aalst, 2008; Weerdt, et al., 2011; Fahland & van der Aalst, 2012). For instance, the output a conformance checking technique may imply that the discovered models perhaps do not describe the executed process as supposed in reality, or is being executed in a different order (Fahland & van der Aalst, 2012; Van der Aalst, 2011). It could also mean that some of the process instance (i.e. individual activities) as observed within the discovered model are skipped in the events log, or may be the logs consist of actions (i.e. events) that are not necessarily defined by the process model (Fahland & van der Aalst, 2012).

Therefore, a well performed *conformance check* is relevant and significant especially from a business objective alignment or auditing perspective. For example, it is possible that the recorded logs could be reiterated (i.e. model replay) against the derived models in order to discover unexpected deviation or bottlenecks that may impact the entire business process in general. In other words, the conformance checking could be utilized to measure the fitness of the models discovered by the PM tools. For instance, the level or extent of behaviours within the event logs which happen to be actually possible according to the discovered process models, and could also be used to perform the repairing of the models in reality. In fact, the conformance checking technique is utilised to balance between traces (i.e. observed behaviours or patterns within the events logs and models) that are *overfitting* or *underfitting* the actual process as performed in reality (Carmona, et al., 2016; Fahland & van der Aalst, 2012). According to (Van der Aalst, 2016), most often conformance check is performed to show the replaying semantics (or better still - token replay) for models with regards to the four quality criteria's - *Fitness*, *Generalisation*, *Precision*, and *Simplicity* (Van der Aalst, 2011).

In summary, the process mining plays an important role in many organisations. It spans its technical application from the fields of data science and business process management (BPM), and as such, we assume that to perform any process mining task that there has to be some kind of recorded data from an actual process. For instance, as this study uses data from the IEEE CIS Task Force on Process Mining (Carmona, et al., 2016) to perform the process models discovery and individual traces classification: which are explained in details in the subsequent sections of this paper. Also, using the Learning Process domain for example, the Figure 1 shows that the first step (i.e. starting point) for any given process mining project is to capture the event data logs about the process, and then generate process model

to show in details how the activities has been performed and to reveal interesting connections between the different process elements (i.e the process instances). In turn, the process mappings can subsequently be utilized to provide methods that allows for an enhanced analysis and/or extension of the discovered process model. Thus, the three types of the process mining techniques – Process Discovery, Conformance Check, and Model Enhancement.

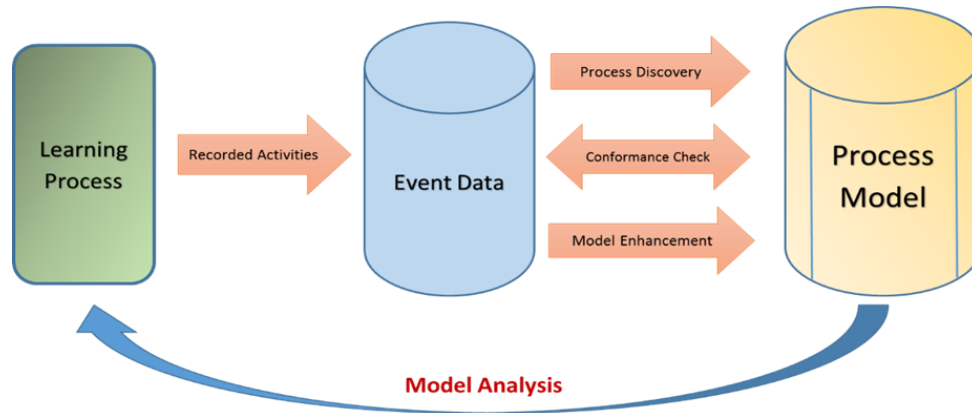


Figure 1. Application of the process mining techniques

USE CASE SCENARIO AND PROBLEM STATEMENT

The proposed Fuzzy-BPMN miner approach in this paper is directed towards the discovery of process models from a set of *Training Event Logs* representing 10 different real-time Business Process executions, and then followed by cross-validation of the derived models with a set of *Test Event Logs* used for evaluation of the process discovery technique and the accuracy of the classification method. Each of the *test event logs* precisely ((test_log_april_1 to test_log_april_10) and (test_log_may_1 to test_log_may_10)) which can be found in (Carmona, et al., 2016) represents part of the original model as recorded by the IEEE CIS Task Force on Process Mining. Also, the *test logs* with complete total of 20 traces for each log are considered to consist of 10 traces which are replayable (i.e. allowed) and another 10 traces which are not replayable (disallowed) by the model. Therefore, the total number of traces for the test event logs (i.e. April log and May log) is thus:

10 test logs x 20 traces which equals to a total number of = *200 Traces* for each of the *April log* and *May log* respectively

Clearly, the aim of the work carried out in this paper is to perform a classification task in order to determine the individual traces that makes up the *test event logs*, and then provide process models with high fitness levels using the Business Process Model Notation (BPMN) mapping for the *training event logs* which allows for testing and evaluation of the classified traces (i.e. the discovered patterns) observed in the test logs. In other words, the objective of the proposed approach is to discover and provide process models that matches the original process models in term of balancing between “overfitting” and “underfitting”. On the one hand, a process model is seen as overfitting (the event log) if it is too restrictive, disallowing behaviour which is part of the underlying process. On the other hand, it is underfitting (the reality) if it is not restrictive enough, allowing behaviour which is not part of the underlying process.

Therefore, following the problem statement and objectives, this article focuses on providing process models which is as good in balancing between “overfitting” and “underfitting” as it is able to correctly classify the traces that can be replayed on the model or not replayable based on the analysis of the classification results.

Thus,

- Given a trace (t) representing real process behaviour, the process model (m) classifies it as allowed, or
- Given a trace (t) representing a behaviour not related to the process, the process model (m) classifies it as disallowed (Carmona, et al., 2016)

In summary, the work in this paper covers the classification attempts for the events logs provided in (Carmona, et al., 2016) and discusses the replaying semantics of the process modelling notation that has been employed. Hence, we reveal how given any process trace (t) (for the test event log) and process model (m) (for the training log) in the employed Fuzzy mining and BPMN notation, it can be unambiguously determined whether or not trace (t) can be replayed on model (m). The study also provides a description of the tools used to discover the process models as well as in checking the result of the classification task. In fact, the method the work has utilized to resolve the identified problem and challenge is grounded on the proposed *Fuzzy-BPMN mining approach* and *Algorithm* as described in the following section.

FUZZY-BPMN MINING APPROACH: METHOD, ALGORITHMS & THE CLASSIFICATION TASK

This section of the paper describes the proposed algorithm and method the work have used to perform the classification task of the Event Data Logs using the Fuzzy-BPMN approach. The method is applied in order to generate the individual traces that makes up each of the process executions as described in (Carmona et al., 2016). In addition, we show how we implement the proposed approach using PM tools such as the Disco (Rozinat & Gunther, 2012) and ProM (Verbeek, et al., 2011; Verbeek, 2014). It is important to note that the proposed algorithm and the defined method of this paper is independent on which tool one may choose in order to analyse the available datasets. Moreover, the work has carefully assessed both by hand and in an automatic manner the performance of the proposed system particularly for comparison and validation purposes.

For instance, the work has used the Disco tool based on the Fuzzy Miner algorithm to generate and map the process models (from the event logs) for conformance checking and further analysis: which allows us to automatically determine the individual *Cases* i.e. the classified traces (20 for each Log) and the sequence of activity executions as performed within the process in reality. And, on the other hand, has carefully cross-validated the results of the classification task (see: Table 1) against the resultant BPMN models that were derived from the training logs. Indeed, the procedures are all aimed at ascertaining the level of performance and accuracy of the classification results of the proposed Fuzzy-BPMN mining approach, particularly in terms of the individual traces and the discovered models. In turn, the following *Algorithm 1* describes how this work discovers and generates (i) *process models* and (ii) *individual traces* - from any event data log containing a *Training sets* and *Test Logs* respectively.

Algorithm 1: Discovering Process Models from Event Logs & Traces Classification

- 1: For all Recorded and Captured Event Data Log *EDL*
- 2: **Input:** *PM* – Process mining tool used to extract model, *M*
 e – Classifier for the event logs, *EDL* and traces, *T*
 case_id(e) - i.e. the Case associated to any event, *e* within the *EDL*
 act_name(e)- i.e. the Activity associated to event, *e* within the *EDL*
- 3: **Output:** Process maps for the discovered models, *M* & individuals traces, *T*
 classifications for the event log, *EDL*

```

4: Procedure: Produce Models,  $M$  from Training Set,  $TSL$  and Traces,  $T$  from
   Test Log,  $TEL$  for cross-validation to determine the model
   traceFitness,  $TF$ 
5: Begin
6: For all Event Data Log  $EDL$ 
7: Extract Process Maps,  $M$ , & Traces,  $T \leftarrow$  from Training Set,  $TSL$  & Test Log,
    $TEL$ 
8: while no more process element is left do
9: Analyze Model,  $M$  and Traces,  $T$  to determine individual tracesFitness,  $TF$ 
10: If  $T \leftarrow$  Null then
11: obtain the occurring activities  $act\_name(e)$  sequence sets from test
   log,  $TEL$ 
12: Else If  $T \leftarrow 1$  then
13: cross-validate resulting Trace,  $T$  from  $TEL$  with discovered Model,  $M$  from
    $TSL$ 
14: If trace,  $T$  exist then
15: For each event Classifier,  $e$  output  $\leftarrow$  return as True_Positive,  $TP$ 
16: Else If trace,  $T$  does not exist then return event Classifier,  $e$  output as
   True_Negative,  $TN$ 
17: Record the traceFitness,  $TF$  in Table as True or False: where each individual
   cell indicates if the discovered model classifies the corresponding trace as
   fitting (allowed i.e.  $TP$ ) or not fitting (disallowed i.e.  $TN$ ).
18: Return: Classification Results of the Experiment and Process Mining
   approach
19: End If statements
20: End while
21: End For

```

Ultimately, from the proposed *Algorithm 1*, we recognises that:

- A typical process model, M consist of Traces, T (i.e. Cases)
- A Trace (Case), T , consist of events, e , such that each event relates to precisely one case.
- Events, e , within a Trace are ordered, most often in a sequential order
- Events for any process mining task must have atleast a Case identification ID (i.e $Case_id$) and Activity Name (i.e Act_name) attributes to allow for the process model discovery.
- Other additional information may be required for ample implementation of the process mining technique e.g. Event ID, Timestamp, Resources, Cost, Roles, Places etc.

Accordingly, the event log that have been provided by the IEEE CIS Task Force on Process Mining (Carmona, et al., 2016) for the models discovery process contains the typical information needed for process mining – particularly in achieving the focus of this article in terms of the process models discovery and implementation of the proposed *Algorithm 1*. The provided *Datasets* represents and shows events logs generated from a business process model to show different behavioural characteristics. We assume that each of the events log contains data related to at least a single process which also refers to a single process instance (i.e. *Case*) and can be related to some *Activity*. Moreover, according to (Van der Aalst, 2011) a “Case ID” and “Activity” is the minimum requirement for any process mining technique. Indeed, the given event logs in (Carmona, et al., 2016) contains the two attributes - *case_id* and *act_name* which precisely specify the requirements that allows for implementation of the proposed process discovery technique in this paper, especially in line with the definition 4.1 in (Van der Aalst, 2011).

Therefore, we assume the following standard:

- $\#case_id(e)$ is the Case associated to any event e .
- $\#act_name(e)$ is the Activity associated to event e .

These definitions are necessary because for the Fuzzy-BPMN miner approach - the activities play an important role in terms of the discovered models, and thus, is used to check for the corresponding cases (i.e. classified traces) within the models. Even more, as there are multiple events referring to the same Activity, we support the filtering of the 200 individual traces (each for the April and May logs) that makes up the test event logs with a *classifier* as described in definition 4.2 in (Van der Aalst, 2011). According to (Van der Aalst, 2011) a classifier is a function that maps the attributes of an event onto a label used in the resulting process model.

Thus, if we use the notation \underline{e} to refer to the events name within the process model, then the classifier for any event in the given *Log* will be, $e \in \mathcal{E}$, where \underline{e} is the name of the event.

More so, since the events are simply identified by their activity name (*act_name*), we then assume:

$$\underline{e} = \#act_name(e)$$

Finally, we apply the classification conversion of the event logs provided (i.e. Simple Event Log, see: Definition 4.4) in (Van der Aalst, 2011) to obtain the individual Log traces.

Therefore, applying the described simple event log definition: Let A be a set of *act_name*. A simple/single trace σ is a sequence of activities, i.e., $\sigma \in A^*$. In other words, a simple event Log, L , is a multiset of traces over some set A .

$$\text{Thus, } L \in \mathbb{B}(A^*).$$

On the other hand, for the *Training Log* there are 1000 cases (trace) that defines the log. However, our focus is to identify the sets of traces (i.e. 200 for *April* and 200 for *May* logs respectively) that characterize the *Test Log* for use in validation of the process model discovery method in this paper, particularly the *April Logs* which were used to score the number of correctly classified traces as well as the experimental outcomes.

Therefore, If we Let $L \subseteq C$ be the event log for the *Test Logs*, and assuming that the classifier $e \in \mathcal{E}$, is applied to the set of sequences of the activities, then from the definition (4.5) in (Van der Aalst, 2011)

$$\langle e1, e2, \dots, en \rangle = \langle e1, e2, \dots, en \rangle$$

where $\underline{L} = [(\hat{c}) \mid c \in L]$ is a simple event log corresponding to *Test Log*.

All the Cases in the *Test Log* are converted into sequences of the activities (*act_name*) using the classifier. Hence

- A Case $c \in L$, is an identifier from the case C .
- $\hat{c} = \#trace(c) = \langle e1, e2, \dots, en \rangle \in \mathcal{E}^*$ is the sequence of events executed for c
- $(\hat{c}) = \langle e1, e2, \dots, en \rangle$ maps these events onto the activity names(*act_name*) using the classifier.

Thus, from the described classification method: ($\underline{e} = \#act_name(e)$), we obtain from the *Log* containing the set of 200 traces for the Test Event Log (test_log_april_1) to (test_log_april_10), i.e., 20 Traces for each log as follows:

$$\begin{aligned} \underline{L}(\text{test_log_april_1}) = & \\ & [\langle b, g, e, q, h, i, l, r, m, o, d, f, p \rangle, \\ & \langle b, b, c, n, h, e, i, q, r, l, m, f, o, d, p \rangle, \\ & \langle g, h, l, q, q, m, r, o, e, d, p \rangle, \\ & \langle j, a, k, b, b, g, e, h, q, l, r, i, m, d, f, o, p \rangle, \\ & \langle b, g, h, i, q, i, r, m, o, d, p, f \rangle, \\ & \langle e, e, e, q, h, r, d, o, r, p \rangle, \\ & \langle g, h, e, i, i, q, l, m, o, f, p, d \rangle, \end{aligned}$$

$\langle b, a, j, k, g, e, q, h, l, i, r, m, o, f, d, p \rangle,$
 $\langle g, i, e, r, l, i, m, d, o, p, d, p \rangle,$
 $\langle b, b, g, e, l, l, h, q, r, r, r, d, o, o, p, f \rangle,$
 $\langle b, g, e, h, i, q, l, r, m, d, p, o, f \rangle,$
 $\langle b, q, g, h, i, h, l, m, m, r, p, f \rangle,$
 $\langle h, g, h, e, r, l, q, i, f, f, p \rangle,$
 $\langle b, j, a, k, g, q, e, i, h, l, r, f, d, o, p \rangle,$
 $\langle c, n, q, e, i, h, r, d, m, o, p, f, p \rangle,$
 $\langle b, g, h, i, e, q, r, l, m, d, o, p, f \rangle,$
 $\langle g, i, h, e, r, q, m, l, o, d, f, p \rangle,$
 $\langle k, b, n, n, c, h, h, e, q, l, q, r, r, i, m, f, f, i, p \rangle,$
 $\langle b, b, b, g, q, i, h, e, r, l, m, f, o, d, p \rangle,$
 $\langle b, b, g, q, e, h, i, r, m, l, d, o, p, f \rangle]$

The Log; \underline{L} (*test_log_april_1*) is example of the set of 20 traces which the work obtained for the *test_log_april_1*. Further details of all the classified traces for the complete *test logs* can be found in (Okoye, et al., 2016).

EXPERIMENTAL SET-UP AND CONFORMANCE CHECKING OF THE INDIVIDUAL CLASSIFIED TRACES

The *Event Logs* used for the process models discovery in this article has been provided in XES (Extensible Event Streams) format. A typical XES document contains logs which consist of *traces*. Each trace describes a sequential list of events corresponding to a particular case in terms of the *concept:name*, for instance, the *case_id* and *act_name* attributes.

The XES files refers to these extensions to provide semantics for the Logs. Truly, in recent times the most widely standard for storing and exchanging event logs across different platforms for process mining is the XES; because the format is less restrictive and truly extendible.

Interestingly, XES has been adopted by the IEEE CIS Task Force on Process Mining since 2010 as standard format for process mining and is supported by tools such as the ProM (Verbeek, et al., 2011; Verbeek, 2014), Disco (Rozinat &

Gunther, 2012), XE-Same (Verbeek, et al., 2011), OpenXES (Gunther & Verbeek, 2014) etc.

Furthermore, a typical attribute for the XES format can be of five core types namely: *String*, *Date*, *Int*, *Float*, and *Boolean*. For instance, the *case_id*, *act_name* which are of *StringType*.

Moreover, these extensions gives semantics for a particular attribute. For example, the extensions corresponds to the *#case_id(e)* and *#act_name(e)* attribute which we used to classify the traces for the test logs.

In fact, there are three classifiers defined by XES which are as follows:

- (i) Classifier Activity (*concept:name*),
- (ii) Classifier Resource (*org:resource*),
- (iii) Classifier Both (*concept:name* and *org:resource*).

Nonetheless, for the purpose of the method and experimentations carried in this paper: our focus is on the *Classifier Activity* because the main objective is to classify the events in the *test logs* based on the *concept:name* attributes i.e. *act_name*, and *case_id* for each of the *Event Name* in terms of their *string values* and order/sequence of the *Lifecycle transition* (as shown in the highlighted part of the Figure 2) and then cross validate the resulting traces with the training set (i.e. the discovered models).

Indeed, XES supports the classifier concept and as such helps in specifying the list of the attributes associated with the *concept:name* as gathered in Figure 2.

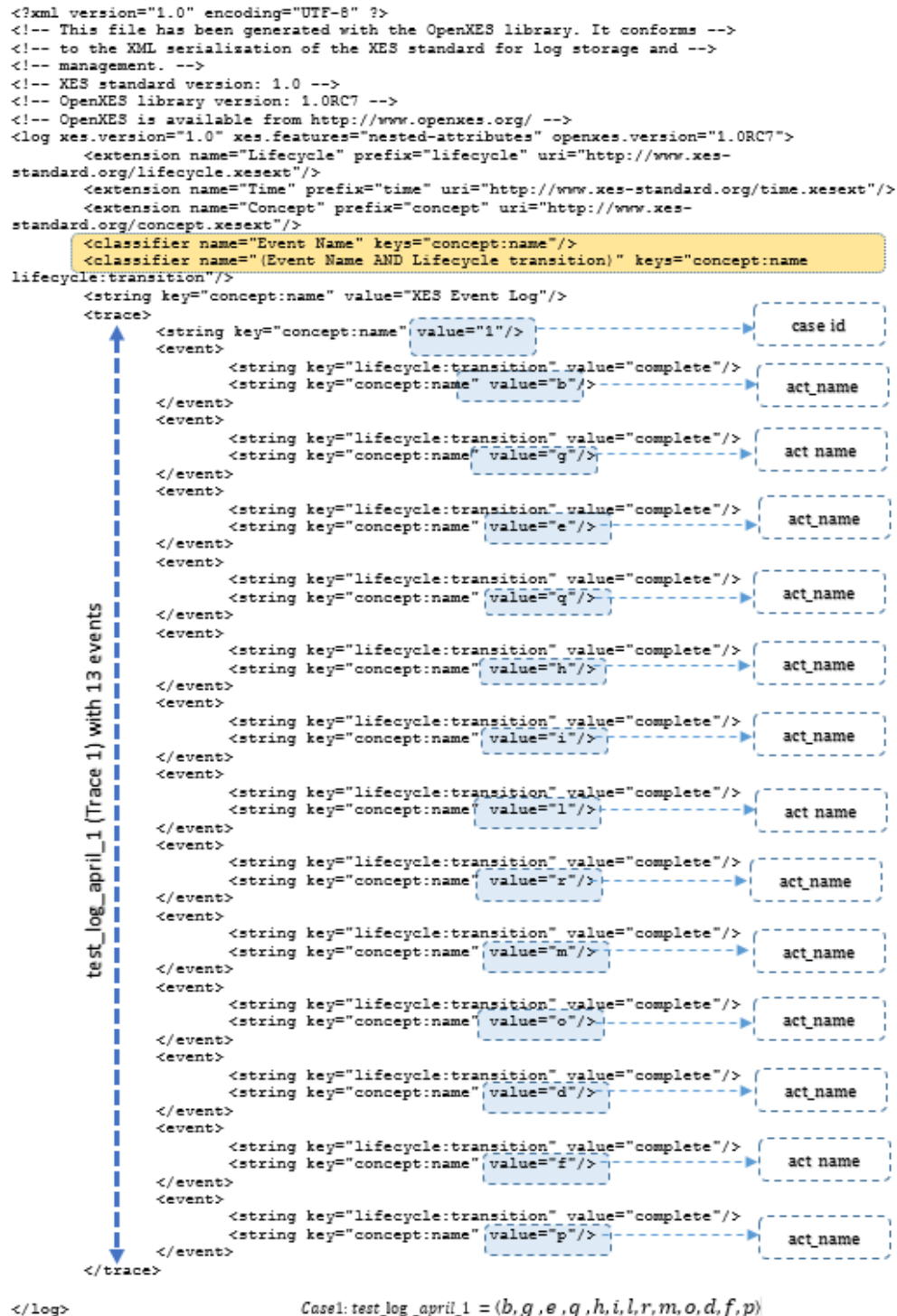
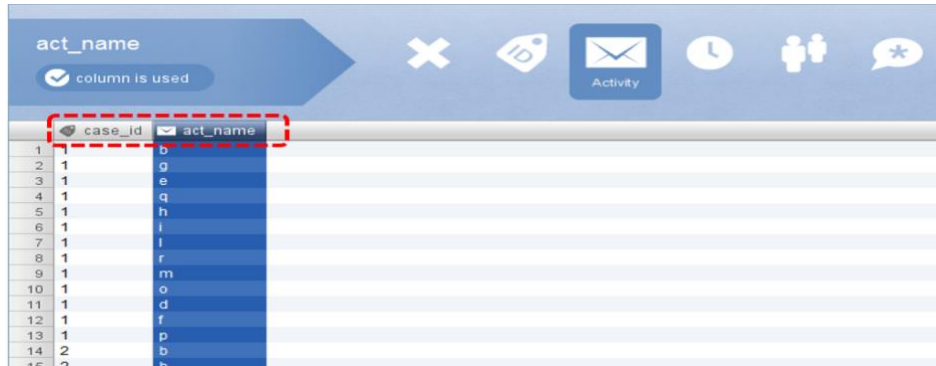


Figure 2. Fragment of the XES file format for the test event log

Following the definitions in the above section and Figure 2, we classified the test event logs. More so, we imported the XES files for the *Test Logs* into Disco (Rozinat & Gunther, 2012) as shown in Figure 3 to see in details how those processes has been performed (i.e. the Process mapping), and more importantly to determine the individual Cases (trace) that makes up the process in order to check if it matches with the classified traces.



	case_id	act_name
1	1	b
2	1	g
3	1	e
4	1	q
5	1	h
6	1	i
7	1	l
8	1	r
9	1	m
10	1	o
11	1	d
12	1	f
13	1	p
14	2	b
15	2	h

Figure 3. Event Log analysis using the Fuzzy miner algorithm in Disco.

In Figure 3 we assigned the *ID Tag* to the first column (i.e *case_id*) in order to identify the events, and the *Activity Tag* to the second column (i.e *act_name*) to determine the sets of activity that makes up the process. Apparently, the outcome of the process is a fuzzy model that represents the various *cases* and *activities sequence* mapping for the events log as shown in Figure 4 and 5.

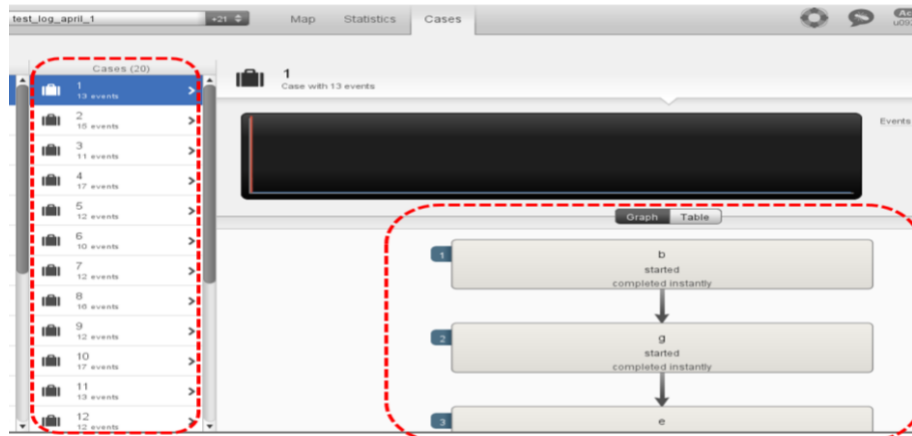


Figure 4. Case View for the *test_log_april_1* showing the 20 cases and graph for the activities sequence.

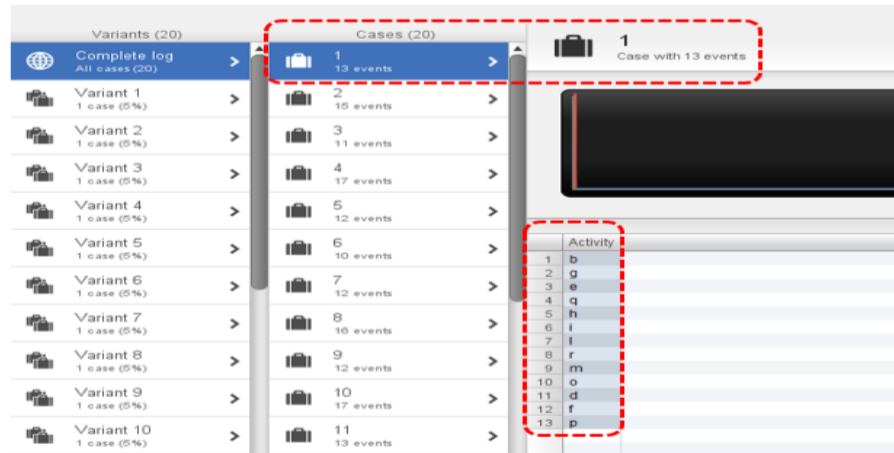


Figure 5. Case view for the *test_log_april_1* showing the 20 cases with an example of case 1 (trace) with 13 events and sets of *Activity* for trace 1.

Indeed, the approach described in Figure 4 and 5 is what we used to check the results of the classification tasks; to see if the outcome of the process confirms to the given event logs.

For example, the activities for the first *case 1* as highlighted in Figure 5, truly corresponds to the first trace discovered by the classifier, i.e.

$\underline{L}(\text{test_log_april_1}) =$

$\{[b, g, e, q, h, i, l, r, m, o, d, f, p], \text{etc.}\}$

To this end, and in view of the individual traces classification results, we make use of the proposed Fuzzy-BPMN mining approach to determine the fitness (replaying semantics) of the individual traces for the *test event logs* by cross-validating the classified traces against the discovered process models from the *training logs* as discussed in the next section of this paper.

THE PROCESS MODELS DISCOVERY METHOD AND ANALYSIS

To discover process models for the event logs (i.e the training logs) used for the experimentations, the work makes use of the Fuzzy miner algorithm in Disco (Rozinat & Gunther, 2012) to process the data. At first, the work discovers 10 different process models from the *training sets* (Carmona, et al., 2016) using the Fuzzy miner (Günther, 2009; Günther & Van der Aalst, 2007; Rozinat & Gunther, 2012) and then subsequently utilize the Business Process Modeling Notation (BPMN) (Van der Aalst, 2016) to analyze and provide the replaying semantics for

the process models. Figure 6 is an example of the discovered models for the *training_log_1* that was analyzed. Further details about the 10 different process models that were discovered and analyzed using the proposed method is described in (Okoye, et al., 2016) and are provided in the *Appendix A* section of this article.

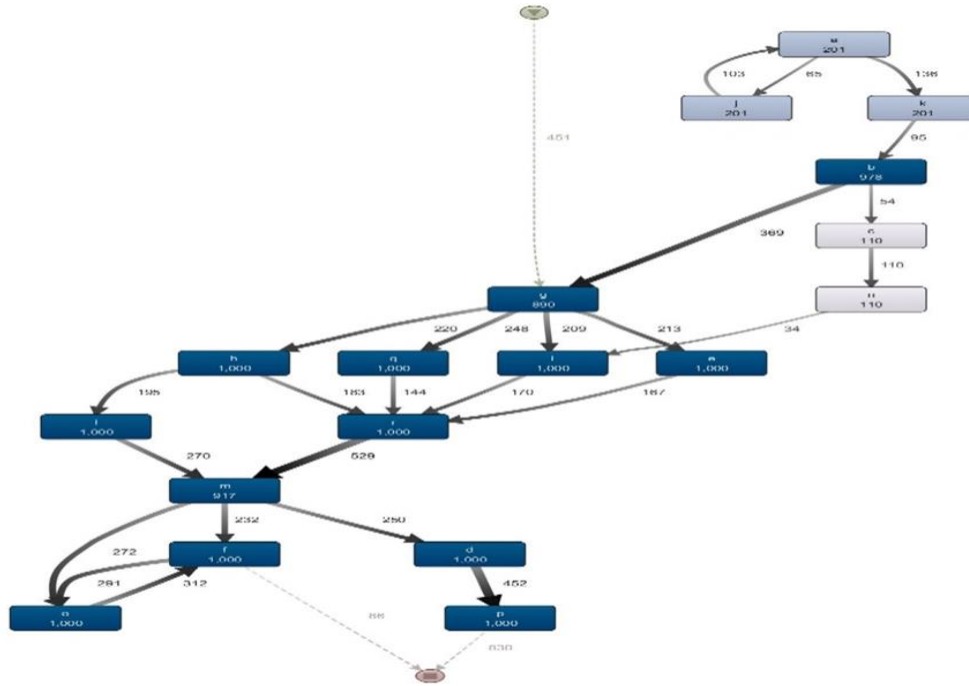


Figure 6. Fuzzy Model discovered for the Training_Log_1

CLASSIFIED TRACES REPLAY AND MODEL FITNESS CALCULATION

Process Mining aims to address the problem of establishing a direct connection between discovered models and the actual low-level event data about the processes in view. Besides, the process discovery techniques allows for viewing the same reality from different angles and at different levels of abstraction. To evaluate and cross-validate the classification tasks for the test event log (i.e *April Log*) with the training model, we base our technique towards balancing between *overfitting* and *underfitting* models as described in section 5.4.3 in (Van der Aalst, 2011) - which focuses on expending measures of data performance indicator using the four quality criterias: *Fitness*, *Precision*, *Generalisation* and *Simplicity* as shown in Figure 7.

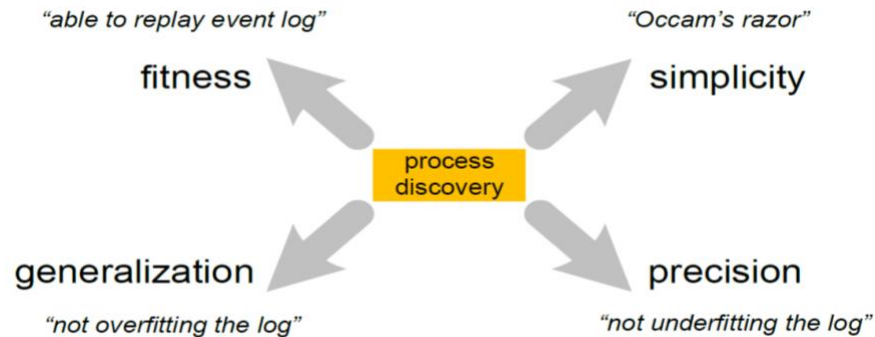


Figure 7. Four competing quality criteria for evaluation of process models (Van der Aalst, 2011)

As gathered in Figure 7, we consider the four quality criteria to explain the level of accuracy (particularly fitness) of the discovered models as defined in section 3.6 in (Van der Aalst, 2011) in order to determine which fractions of the traces in the *test logs* can be fully replayed or are disallowed by the discovered models.

Thus:

- *Fitness*: the discovered model should allow for behaviours seen in the event log. Thus, is the event log possible according to the discovered model?
- *Precision (avoiding underfitting)*: the discovered model should not allow for behaviours completely unrelated to what was seen in the event logs. Thus, is the model not underfitting i.e. allows for too much?
- *Generalization (avoiding overfitting)*: the discovered model should generalize the example behaviours seen in the event logs. Thus, is the model not overfitting i.e. only allows for particular examples?
- *Simplicity (Occam's razor principle)*: the discovered model should be as simple as possible. Thus, is the discovered model the simplest? One should not increase, beyond what is necessary, the number of entities required to explain anything, i.e., one should look for the "simplest model" that can explain what is observed in the dataset.

Essentially, the fitness of the discovered models is judged on the *Training Logs* which are measured against the *test logs* classification results as shown in Figure 8 - also referred to as *Cross-Validation* in section 3.6.2 in (Van der Aalst, 2011).

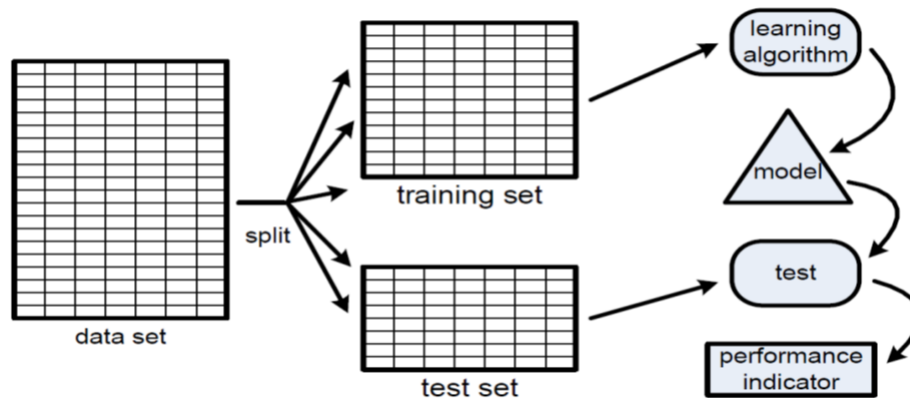


Figure 8. Cross-validation using a training set and test set (Van der Aalst, 2011)

Furthermore, according to (Van der Aalst, 2011), the conformance checking is closely related to measuring the fitness of the discovered models, and it can also be used to evaluate and compare the process discovery algorithms. Section 7.2 of (Van der Aalst, 2011) discusses the replaying semantics (*Token Replay*) for the process models with respect to the four quality criteria. The token replay shows how the notion of event log fitness can be quantified i.e. the proportion of behaviours in the event logs that are possible according to the discovered models. In other words, the token replay are used to establish a tight coupling between the *discovered model* and the *event logs*.

For that reason, to achieve the set objective of the paper - it was necessary to construct BPMN models with notational elements (as explained in Figure 9) capable of describing the nesting of individual activities (*traces*) by using the event-based split and join gateways, i.e. *AND*, *XOR*, and *OR* etc. Moreover, since our target is to classify as correctly as possible the traces which are allowed and the traces which are not allowed in the original model, the work utilized the BPMN event-based gateways to replay the individual traces fitness alongside the derived models from the training event logs, and in so doing, identify which traces that are fitting or not fitting the original model.

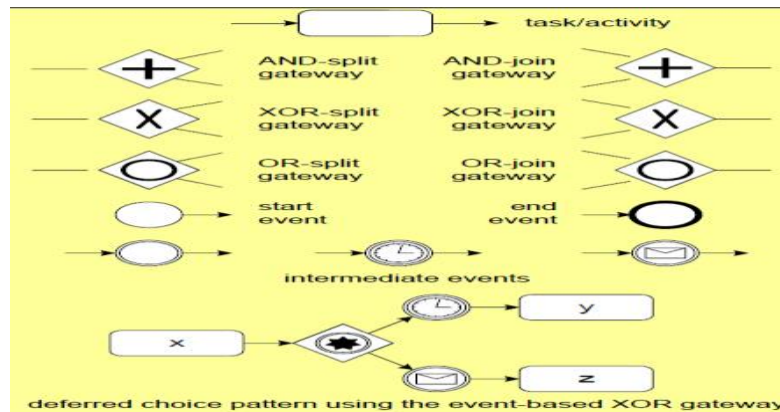


Figure 9. BPMN Gateway with Notational elements (Van der Aalst, 2011)

Indeed, an event in a BPMN model can be compared to a place within a Petri-net (Van der Aalst, 2011), and just like Petri nets, are token based semantics which can be used to replay a particular trace within a discovered process model (Van der Aalst, 2016). To this end, the work makes use of the *Convert Petri net to BPMN* plugin in ProM (Verbeek, et al., 2011) to discover the BPMN models for the training logs. Figure 10 is an example of the discovered BPMN Diagram for the *training_log_1*. Further details about the other 10 different BPMN models that was discovered using the method can be found in (Okoye, et al., 2016) and also included in the *Appendix B* section of this paper.

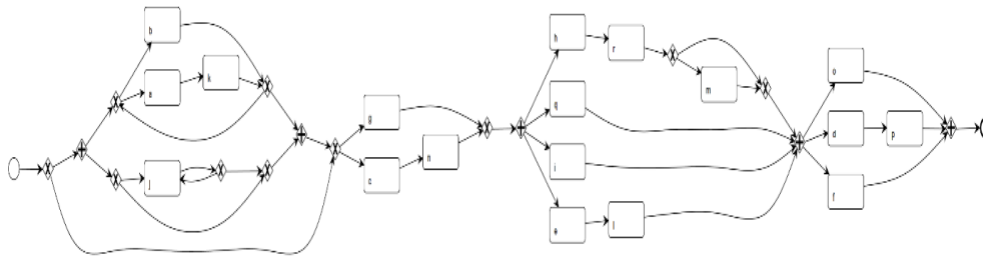


Figure 10. Example of BPMN model discovered for the training_log_1

Consequently, in Table 1 the study presents the classification results of the Fuzzy-BPMN mining approach for the *test event logs* cross-validated against the corresponding *training set* (model): where each individual cell indicates if the discovered model classifies the corresponding trace as fitting (i.e allowed) or not fitting (disallowed). Thus, the *columns* represents the process models for the 10 training logs, while the *rows* represents the individual traces for the test log. For example, cell at row “Trace_3” column “Training model_5” contains the classification attempt for the 3rd trace discovered from the test_log_april_5 cross-validated against the training_log_5.

Table 1. Classified Trace fitness Table for the test event logs (*test_log_april_1* to *test_log_april_10*)

	Training model_1	Training model_2	Training model_3	Training model_4	Training model_5	Training model_6	Training model_7	Training model_8	Training model_9	Training model_10
Trace_1	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_2	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_3	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_4	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
Trace_5	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	TRUE
Trace_6	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_7	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_8	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE
Trace_9	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_10	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_11	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_12	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_13	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_14	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Trace_15	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_16	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
Trace_17	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE
Trace_18	FALSE	TRUE	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE
Trace_19	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE
Trace_20	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

RESULTS AND OUTCOME OF THE FUZZY-BPMN MINING APPROACH

The IEEE CIS Task Force on Process Mining contest committee published on the website (Carmona, et al., 2016) - (a) 10 test logs, each of which contains 20 traces that were used to score the submission report, and (b) 10 reference process models in BPMN generated from the original event logs which were not previously revealed. The Table 2 represents the final results and scoring of the employed Fuzzy-BPMN mining approach and experimentations in this paper.

Table 2. Trace Fitness and Classification Table for the Test Event Logs (test_log_april_1 to test_log_april_10) using the Fuzzy-BPMN Miner

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8	Model 9	Model 10
Trace_1	TP *	TN *	TP *	FP	TN *	FP	TP *	TP *	TP *	TP *
Trace_2	TN *	TN *	TP *	TP *	TP *	TP *	TP *	TN *	TP *	TP *
Trace_3	TP *	TP *	TP *	TN *	TN *	FP	FP	TP *	TP *	TN *
Trace_4	TP *	TP *	FP	TP *	TN *	TP *	TN *	TP *	TP *	FP
Trace_5	TN *	FP	FP	TP *	TN *	TP *	TN *	TP *	TP *	TN *
Trace_6	TP *	FP	FP	TP *	TN *	TP *	TP *	TN *	TN *	TP *
Trace_7	TN *	TP *	TP *	TN *	TN *	TP *	TN *	TP *	TP *	TN *
Trace_8	TN *	TP *	TP *	FN	TN *	FP	TP *	TP *	TP *	TP *
Trace_9	TP *	TN *	TP *	TN *	TP *	FP	TP *	TP *	TN *	TP *
Trace_10	TP *	FP	TP *	TN *	TN *	FP	TP *	TP *	TP *	TP *
Trace_11	TN *	TP *	TP *	FN	TP *	TN *	TN *	FP	TN *	TP *
Trace_12	TP *	FP	FP	TP *	TP *	TP *	TP *	FP	TP *	TN *
Trace_13	TP *	TP *	FP	TN *	TP *	FP	TN *	TN *	TN *	TP *
Trace_14	TN *	TP *	TN *	TN *	TN *	FP	TN *	TP *	TN *	TP *
Trace_15	TP *	TN *	TN *	TN *	TP *	TP *	TN *	TN *	TN *	TN *
Trace_16	TN *	TN *	FP	TP *	TP *	FP	TN *	FP	TP *	TN *
Trace_17	TP *	TP *	TP *	TP *	TP *	TP *	TP *	TN *	TN *	TP *
Trace_18	TN *	TP *	FP	TN *	TP *	TP *	TP *	TN *	TN *	TN *
Trace_19	TN *	TP *	TP *	TP *	TN *	TP *	TP *	TP *	TN *	TN *
Trace_20	TN *	TN *	FP	TN *	TP *	FP	TN *	TN *	TP *	TN *

True Positive (TP) :	10	10	10	8	10	10	10	10	10	10
False Positive (FP):	0	4	8	1	0	9	1	3	0	1
True Negative (TN):	10	6	2	9	10	1	9	7	10	9
False Negative (FN):	0	0	0	2	0	0	0	0	0	0
NO. of traces correctly classified	20	16	12	17	20	11	19	17	20	19

The cells colours indicates the classification attempt for each of the traces discovered from the test event logs. Also, the cells with gold sign * indicates the traces that were correctly classified by the Fuzzy-BPMN Miner with total of 171 traces out of 200.

Consequently, as shown in Table 2 the following performance metrics (Van der Aalst, 2011; Van der Aalst, 2016) were used to measure the fitness of the individual traces for the datasets, where:

- ❖ *TP* is the number of *true positives* i.e. instances that are correctly classified as positive
- ❖ *FN* is the number of *false negatives* i.e. instances that are predicted to be negative but should have been classified as positive
- ❖ *FP* is the number of *false positives* i.e. instances that are predicted to be positive but should have been classified as negative
- ❖ *TN* is the number of *true negatives* (i.e. instances that are correctly classified as negative)

Accordingly, the cells with gold sign (*) indicates the traces that were correctly classified by the Fuzzy-BPMN miner after scoring the classification results and models. Indeed, the final result after scoring by the committee experts in process

mining (panel of judges) shows that the Fuzzy-BPMN miner approach has correctly classified 171 out of 200 (85.5%) traces in the original process model.

Presently, the only other contest related to the PM is the annual Business Process Intelligence Challenge (BPIC) (van Dongen, et al., 2016) which makes use of real life datasets, but without an objective evaluation criteria. The BPIC contest focuses more on the observed values of the process mining and analysis techniques, and as such does not limit its submissions to the process discovery methods. For instance, the contest also looks at some performance analysis techniques, conformance checking etc. However, the submissions are also being assessed by a panel of expert judges within the PM field. On the other hand, the BPM Process Discovery Contest (Carmona, et al., 2016) is quite different from the BPIC because it focuses more on the assessment of process discovery techniques. In essence, datasets which are synthetic in nature are used to have an objectified “proper” answer to process mining problems. Thus, the process discovery is turned into a classification task with a training set and a test set; where a discovered process model needs to decide whether the classified ‘traces’ are fitting or not.

DISCUSSIONS & LIMITATIONS

The work in this paper shows that the construction of useful process models and the description of the causal dependencies that exist between the various activities as performed in reality - requires a well performed and fit-for-purpose PM approach. On the whole, one can make use of the amalgamation of different process discovery method (such as the Fuzzy-BPMN miner proposed in this paper, i.e., Hybrid Algorithm) to obtain process models which are as good in balancing between *overfitting* and *underfitting* as it is able to correctly classify the traces that can be replayed (allowed) or non-replayable (disallowed) based on the analysis of the event logs and the discovered models.

In short, the main benefits of the Fuzzy-BPMN mining approach, sets of algorithms and the experimentations carried out in order to address the research questions in this paper can be summarised as follows:

- A process mining technique that is capable to a greater percentage; accurately classify the individual traces (i.e. the process elements or activities) and induce new knowledge based on previously unobserved behaviours within the process knowledge-base.

- A set of process mining algorithm that proves useful towards the discovery, monitoring and enhancement of the analysis of event logs about any domain process or data by discovering useful and worthwhile process models.
- A method that proves useful towards the transformation of events data logs for process mining into minable executable formats to support the discovery process.
- A series of case study and experimentations (using the real life data from the Business Process) showing that the Fuzzy-BPMN miner can be used to enhance the classification process of any given events log as well as the discovered process models and their analysis.

Indeed, to achieve the stated contributions of the paper, the work assesses the level of accuracy of the classification results of the Fuzzy-BPMN miner to predict behaviours of unobserved traces (i.e process instances) within the process-base by determining which traces are fitting (true positives) or not fitting (true negatives) the discovered models - using the *training sets* and *test logs* from the IEEE CIS Task Force on Process Mining (Carmona et al., 2016) for the cross-validation experiment. Moreover, the proposed Fuzzy-BPMN approach could be regarded as a fusion theory that is based on the fuzzy logics and devoted to represent and analyse information at the *process-levels* rather than the *data-levels*. Apparently, the fuzzy logic (Zadeh, 1999) has since been introduced as an extension of the Boolean logic which allows a proposal to be in another state as true or false (Dammak, et al., 2014) by enabling the modelling of uncertainty and imprecision that often characterize the human representations of knowledge and/or the captured datasets.

Furthermore, owing to the fact that the Fuzzy miner algorithms are practically used to discover process models in a more or less precise way and to visualize complex processes, the work makes use of the combination of the Fuzzy and BPMN miner (independent on which tool or platform that it is being utilized or used in e.g. the Disco or ProM) to analyse the available datasets. In other words, flexible and more or less structured models (Rozinat, 2010; Günther, 2009). According to (Rozinat, 2010) fuzzy miner algorithms are applied with the goal to show understandable models for very unstructured processes. Even the author in (Ingvaldsen, 2011) is more specific about the potential benefit of using the fuzzy mining technique, and notes that the fuzzy miner is a one of the many existing algorithms which aims to address the problem of mining complex processes (that are unstructured in nature) by utilizing a mixture of clustering and abstraction methods. This means that models discovered as a result of applying the fuzzy miner algorithm are able to abstract from details and aggregate behaviours that are not of interest (i.e. visual

noise) to the process analysts by grouping the sets of activities into cluster nodes (Rozinat, 2010). Even though, by resolving the unstructured processes and complexities, we mean that the fuzzy miner algorithms are used to produce simplified models to directly address the problems of large numbers of activities and/or highly unstructured datasets or behaviours (Okoye et al., 2017).

Nevertheless, one of the main limitations of the fuzzy miner algorithms is that they tends to lack some kind of formal description (i.e. semantics). For example, the successive pattern recognition that is missing in the discovered models - such as simple choice (i.e. OR split), parallel choice (i.e. AND split), or multiple choice (i.e. XOR split) which can be used to described the casual dependencies or semantics of the various activities as performed in reality. Thus, there are no explicit distinction possible between the events splits and/or join gateways.

To this end, this paper has shown that it is possible to integrate the fuzzy models with other tools in order to overcome the aforementioned limitations. The work uses the integration of the Fuzzy with the BPMN approach to construct process models with notational elements that are capable of describing the nesting of individual activities (process instances) by using the event-based split and join gateways - i.e. AND, XOR, and OR etc. The process is applied as means towards resolving such limitations that are generally related to the fuzzy models: where most often the fuzzy models appears to be relaxed in nature especially when compared with the semantics of other process modelling languages such as the Petri nets or BPMN. In other words, the paper reveals how the events gateways in BPMN model (also referred to as token based semantics) can be used to replay a particular trace within the discovered models (Van der Aalst, 2011; Van der Aalst, 2016) and as such overcomes the identified limitations with the fuzzy models. Thus, the amalgamation and proposal of the Fuzzy-BPMN Miner.

On the other hand, the research proposal and experimentations in this paper have identified and introduce state of the art tools which are suitable for process mining, particularly in relation to the accuracy of the classification process and mining outcomes. Specifically, the paper have proposed a hybrid or combination of PM algorithms that proves to accurately classify to high percentage - the traces that can be found within the event logs and resultant process models. However, whilst the work believes that such methods are practically suitable for effective process models discovery and valuation of the fitness of the derived models, there could also exist a number of limitations and threats to validity. Hence, even though one of the main benefit of the method is that it appears to be a fusion theory which integrates the fuzzy model with other tools; such as the BPMN. In many settings, fuzzy models have proven to be ambiguous and characteristically contains vast

number of arc nodes which are disjointed via impounded nodes that are primitive in nature. Therefore, with such process models, it may not be likely to extract meaningful (semantic) information about the process elements. Although, for that reason, this work has shown that it is possible to improve the information values of such type of models to some greater extent by carefully integrating and tuning the semantics metrics that those models lack through the amalgamation of the Fuzzy miner with the BPMN models - which has proven to be capable of describing the nesting of individual activities (i.e. the semantics of the process elements) by using the event-based split and join gateways. Moreover, the process seems to be a cumbersome task and does not guarantee and/or carry some threats to the validity of the outcomes.

An additional threat to validity of the work in this paper is that there are no currently tools capable of directly converting the fuzzy models into some other modelling formats or notation. As a consequence, the work leverages a varied range of events log conversion in order to achieve different viewpoint about the event logs. Indeed, future works could focus on extending the proposed approach through provision of tools capable of automatically integrating such metrics with the fuzzy models in order to support their analysis at a more abstract level, and better still, guarantee the accuracy of the results. Besides, this work has shown that a way to resolve those problems is to provide the option for specifying semantics which in turn is capable of allowing for an accurate analysis of such models.

Nonetheless, this research believes that there is a lot of opportunities for future works in extending the proposed approach in this paper. Further, a worthwhile extension will be to complement the fuzzy models with a platform for completely automatic discovering and/or integration of the semantic information that those models lack.

Finally, in addition to the aforementioned areas that could be considered for future works, another potentially worthwhile area to pursue in the future is to expound the current system to include and spread out to diverse organisations or business owners in their current business processes or operational settings. This may include the development of authoring tools capable of augmenting the stated achievements of this paper, or yet still, improve the outcome of the classification task that have already been well-defined in this article.

CONCLUSION

This article presents a Fuzzy-BPMN mining approach that makes use of a *training events log* representing 10 different real-time business process executions to provide a method for discovery of useful and worthwhile process models, and then cross-validates the derived models with a set of *test event logs* in order to measure the performance of the proposed method. Thus, we reveal how given any process trace (t) (for the test event log) and process model (m) (for the training log) in the discovered Fuzzy models and BPMN notation, it can be unambiguously determined whether or not trace (t) can be replayed on model (m). In turn, the study provides a description of the tools used to discover the process models as well as in checking the results of the classification tasks; for comparisons and validation purposes. Overall, the work looks at the sophistication of the proposed Fuzzy-BPMN approach in terms of the discovered models, validation of the classification tasks, and the impact of the research experiments and outcomes. Indeed, the results of the classification process (by the proposed Fuzzy-BPMN miner) after review by experts within the process mining field; shows that the Fuzzy-BPMN miner approach has correctly classified 171 out of 200 (85.5%) traces in the original process model. Clearly, the outcome of the process indicates that the Fuzzy-BPMN miner proves to be a useful technique towards the discovery, monitoring and enhancement of the process analysis of event logs about any domain process or data, especially, when compared to other standard logical procedures used for process mining.

Future work will be to implement the proposed approach in order to analyse data from other domain areas of interest. This will allow for further validation and will generalise the findings and valuation of the process mining approach presented in this paper. Another potential extension will be to complement the method with a platform for completely automatic discovering and integration of the fuzzy models with semantic information or knowledge.

REFERENCES

- Adriansyah, A., Van Dongen, B. & Van der Aalst, W. M. P., 2011. *Conformance Checking using Cost-Based Fitness Analysis..* Helsinki, Finland., IEEE International Enterprise Computing Conference (EDOC 2011) IEEE Computer Society.

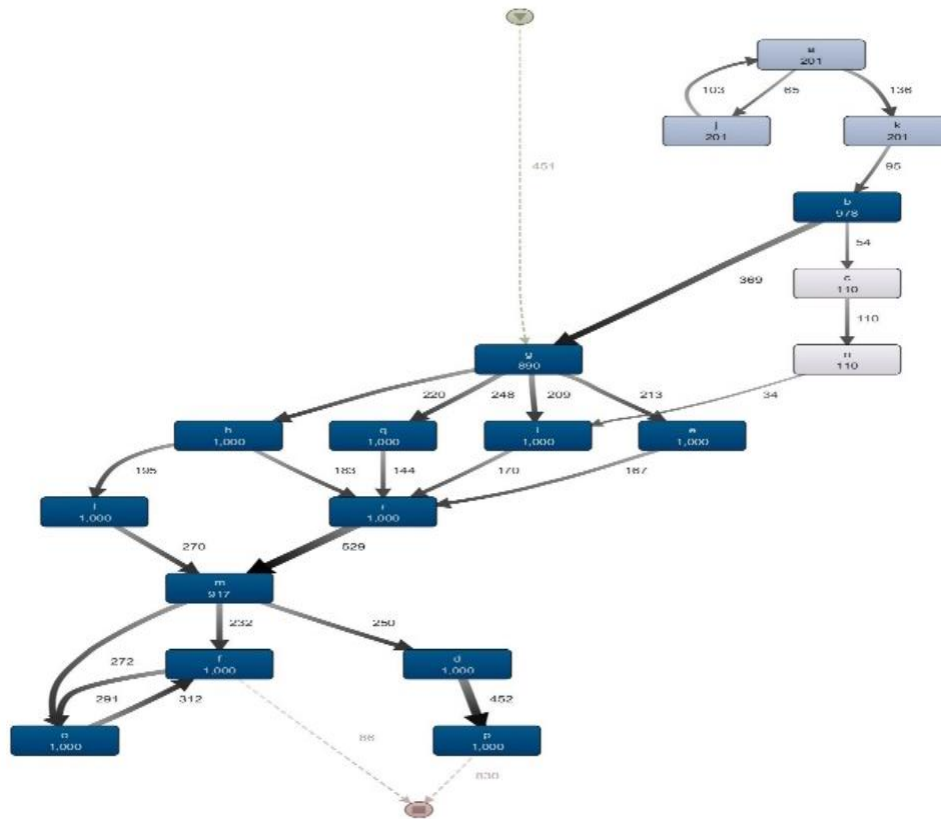
- Cairns, A. H.; Gueni, B.; Fhima, M.; Cairns, A.; David, S.; Khelifa, N., 2015. Process Mining in the Education Domain. *International Journal on Advances in Intelligent Systems*, vol 8(1 & 2).
- Cairns, A. H.; Ondo, J. A.; Gueni, B.; Fhima, M.; Schwarcfeld, M.; Joubert, C. & Khelifa, N. 2014. *Using Semantic Lifting for Improving Educational Process Models Discovery and Analysis*. s.l., SIMPDA, volume 1293 of CEUR Workshop Proceedings, CEUR-WS.org, pp. 150-161.
- Carmona, J., de Leoni, M., Depair, B. & Jouck, T., 2016. *Process Discovery Contest @ BPM 2016*, Ist [ed] Rio de Janeiro: IEEE CIS Task Force on Process Mining.
- Chesani, F., Ciampolini, A., Loreti, D. & Mello, P., 2016. Process Mining Monitoring for Map Reduce Applications in the Cloud. *ACM Digital Library - Proceedings of the 6th International Conference on Cloud Computing and Services Science*, 1 and 2(1), pp. 95-105.
- Dammak, S. M., Jedidi, A. & Bouaziz, R., 2014. *Fuzzy semantic annotation of Web resources*. Sousse, 2014 World Symposium on Computer Applications & Research (WSCAR), pp. 1-6.
- De Leoni, M. & Van der Aalst, W. M. P., 2013. Data-Aware Process Mining: Discovering Decisions in Processes Using Alignments. In: S. Y. Shin & J. C. Maldonado, eds. *ACM Symposium on Applied Computing (SAC 2013)*. Coimbra, Portugal: ACM Press, New York, NY, pp. 1454-1461.
- de Medeiros, A. K. A. & Van der Aalst, W. M. P., 2009. Process Mining towards Semantics. In: T. Dillon, E. Chang, R. Meersman & K. Sycara, eds. *Advances in Web Semantics I. Lecture Notes in Computer Science*, vol 4891. Berlin, Heidelberg: Springer, Berlin, Heidelberg, pp. 35-80
- Dou, D., Wang, H. & Liu, H., 2015. *Semantic Data Mining: A Survey of Ontology-based Approaches*. California, USA, 9th IEEE Int. Conference on Semantic Computing, p. 244 – 251.
- Elhebir, M.H.A & Abraham, A., 2015. A Novel Ensemble Approach to Enhance the Performance of Web Server Logs Classification, *Int. Journal of Computer Information Systems and Industrial Management Applications* ISSN 2150-7988 Vol. 7 (2015) pp. 189-195.
- Fahland, D. & van der Aalst, W. M. P., 2012. Repairing Process Models to Reflect Reality. In: B. A., G. A. & K. E., eds. *Business Process Management. BPM 2012. Lecture Notes in Computer Science*, vol 7481. Berlin, Heidelberg: Springer, Berlin, Heidelberg, pp. 229-245.

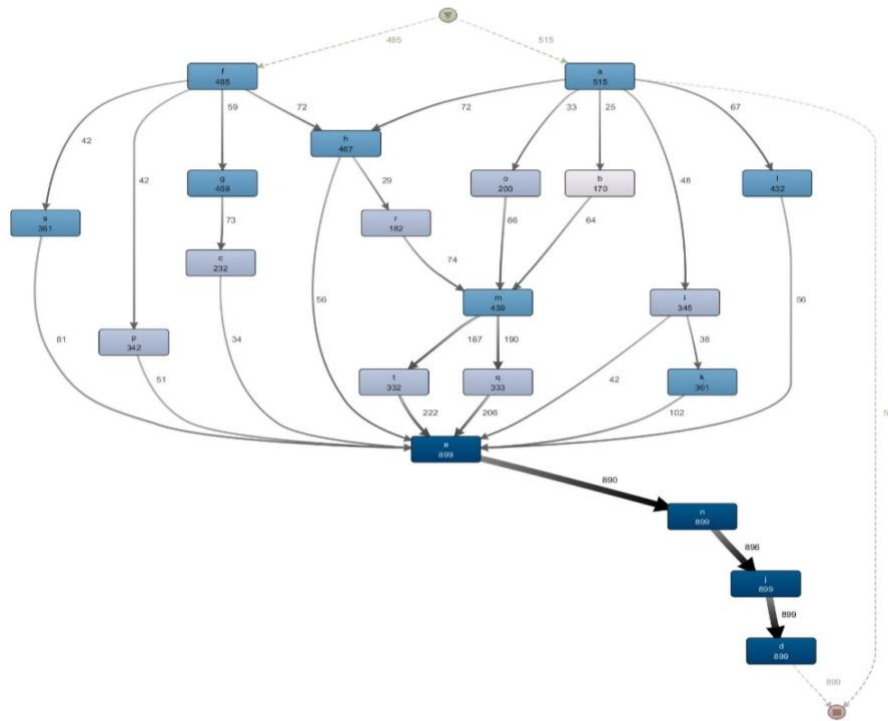
- Greco, G., Guzzo, A., Pontieri, L. & Sacca, D., 2006. Discovering Expressive Process Models by Clustering Log Traces. *IEEE Transaction on Knowledge and Data Engineering*, 18(8), pp. 1010-1027.
- Günther, C. W., 2009. *Process Mining in Flexible Environments.*, Eindhoven, the Netherlands: PhD thesis, Department of Technology Management, Technical University Eindhoven.
- Günther, C. W. & Van der Aalst, W. M. P., 2006. A generic import framework for process event logs. In: J. Eder & S. Dustdar, eds. *Business Process Management Workshops*. Berlin: Springer, Berlin,, pp. 81-92.
- Gunther, C. W. & Verbeek, E., 2014. *OpenXES - Developer Guide 2.0*, Netherlands: IEEE 1849-2016 XES. Available at: <http://www.xes-standard.org/openxes/developerguide> [Accessed 20 May 2017].
- Han, J. & Kamber, M., 2005. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Ingvaldsen, J. E., 2011. *Semantic process mining of enterprise transaction data*, Norway: PhD Thesis - Norwegian University of Science and Technology.
- Jans, M. J., 2011. Process Mining in Auditing: From Current Limitations to Future Challenges. In: D. F., B. K. & D. S., eds. *Lecture Notes in Business Information Processing*. Berlin, Heidelberg: International Conference on Business Process Management Workshops. BPM 2011. Springer,, pp. 394-397.
- Munoz-Gama, J. & Carmona, J., 2011. *Enhancing Precision in Process Conformance: Stability, Confidence and Severity*. Paris, France, Proceedings of CIDM 2011. IEEE.
- Okoye, K., Tawil, A. R. H., Naeem, U. & Lamine, E., 2016. Discovery and Enhancement of Learning Model Analysis through Semantic Process Mining*. *International Journal of Computer Information Systems and Industrial Management Applications*, 8(2016), pp. 093-114.
- Okoye, K., Tawil, A. R. H., Naeem, U. & Lamine, E., 2016. *Fuzzy-BPMN miner approach - Process Discovery Contest @ BPM 2016*, Rio de Janeiro: Technical Report, IEEE CIS Task Force on Process Mining discovery contest [1st Ed] in BPI workshop at BPM 2016 Conference.
- Okoye, K., Naeem, U. & Islam, S., 2017. Semantic Fuzzy Mining: Enhancement of process models and event logs analysis from Syntactic to Conceptual Level. *International Journal of Hybrid Intelligent Systems (IJHIS)*, IOS Press, 14 (1-2), pp. 67-98.

- Rojas, E., Munoz-Gama, J., Sepúlveda, M. & Capurro, D., 2016. Process mining in healthcare: A literature review. *Journal of Biomedical Informatics*, 61(1), pp. 224-236.
- Rozinat, A., 2010. *Process Mining: Conformance and Extension*, Eindhoven, the Netherlands: PhD Thesis. Technische Universiteit Eindhoven.
- Rozinat, A. & Gunther, C., 2012. *Disco User Guide - Process Mining for Professionals*, Eindhoven, The Netherlands: Fluxicon.com.
- Rozinat, A. & Van der Aalst, W. M. P., 2008. Conformance Checking of Processes based on Monitoring Real Behaviour. *Journal of Information Systems*, 33(1), p. 64–95.
- Van der Aalst, W. M. P.; Van Dongen, B. F.; Herbst, J.; Maruster, L.; Schimm, G.; Weijters, A. J., 2003. Workflow Mining: A Survey of Issues and Approaches. *Data and Knowledge Engineering*, 47(2), pp. 237-267.
- Van der Aalst, W. M. P., 2011. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. 1 ed. Berlin: Springer.
- Van der Aalst, W. M. P., 2016. *Process Mining: Data Science in Action*. 2nd ed. Berlin: Springer-Verlag Berlin Heidelberg.
- Van der Aalst, W. M. P., van Hee, K., van Werf, J. M. & Verdonk, M., 2010. Auditing 2.0: Using Process Mining to Support Tomorrow's Auditor. *IEEE Computer*, 43(3), pp. 90-93.
- Van der Aalst, W. M. P., Weijters, A. J. M. M. & Maruster, L., 2004. Workflow Mining: Discovering Process Models from Event Logs.. *International Journal of IEEE transactions on Knowledge and Data engineering*, 16(9), pp. 1128-1142.
- Van der Aalst, W. M. P., Adriansyah, A., de Medeiros, A. K. A. & et al, 2012. Process Mining Manifesto. In: D. F., B. K. & D. S., eds. *Business Process Management Workshops. BPM 2011. Lecture Notes in Business Information Processing*. Berlin: BPM Workshops LNBIP, vol. 99, pp. 169-194. Springer, 2012., pp. 169-194.
- Van Dongen, B., Claes, J., Burattin, A. & De Weerd, J., 2016. *12th International Workshop on Business Process Intelligence 2016*. [Online] Available at: <http://www.win.tue.nl/bpi/doku.php?id=2016:start#organizers> [Accessed 20 June 2017].

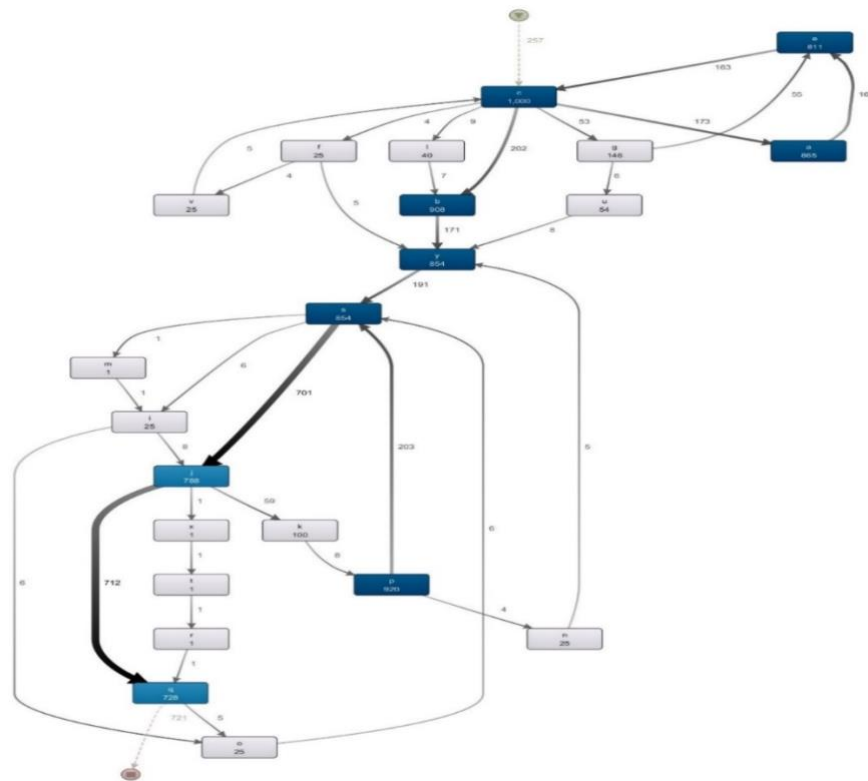
- Verbeek, H., 2014. *Process Mining research tools and application*. [Online] Available at: <http://www.processmining.org/promimport/start> [Accessed 31 October 2016].
- Verbeek, H., Buijs, J., van Dongen, B. & van der Aalst, W. M. P., 2011. XES, XESame, and ProM 6. In: P. E. (. Soffer P., ed. *Information Systems Evolution*. Springer, Berlin, Heidelberg: CAiSE Forum 2010. Lecture Notes in Business Information Processing, pp. 60-75.
- Weerdt, J. D., Backer, M. D., Vanthienen, J. & B., B., 2011. *A Robust F-measure for Evaluating Discovered Process Models*. Paris, France, In Proceedings of CIDM, 2011. IEEE, p. 148–155.
- Zadeh, L. A., 1999. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100(1), pp. 9-34.

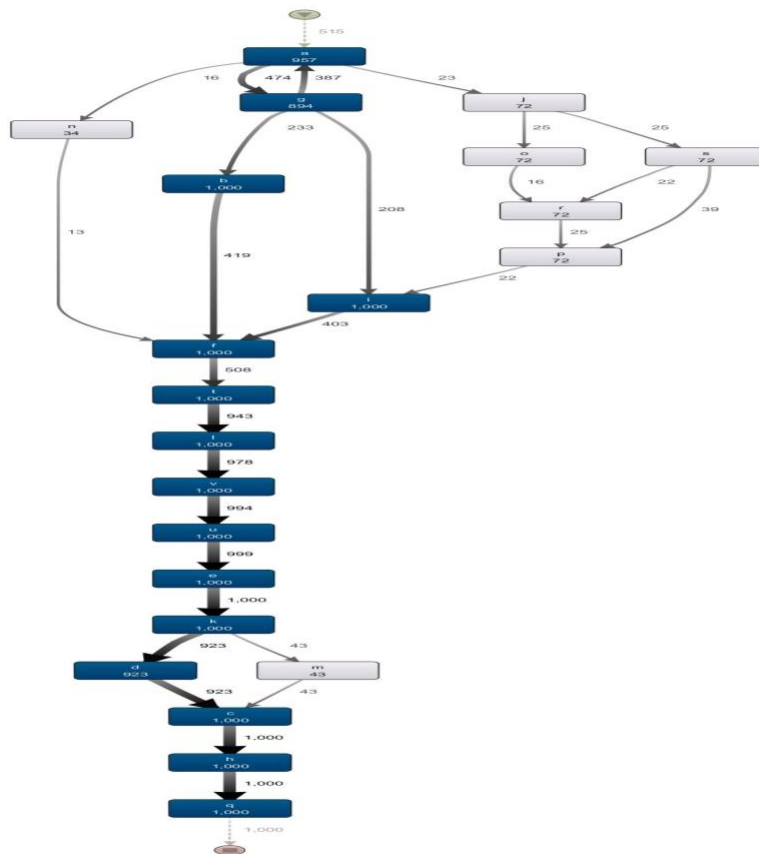
APPENDIX A

A.1 Fuzzy Model for *training_log_1*

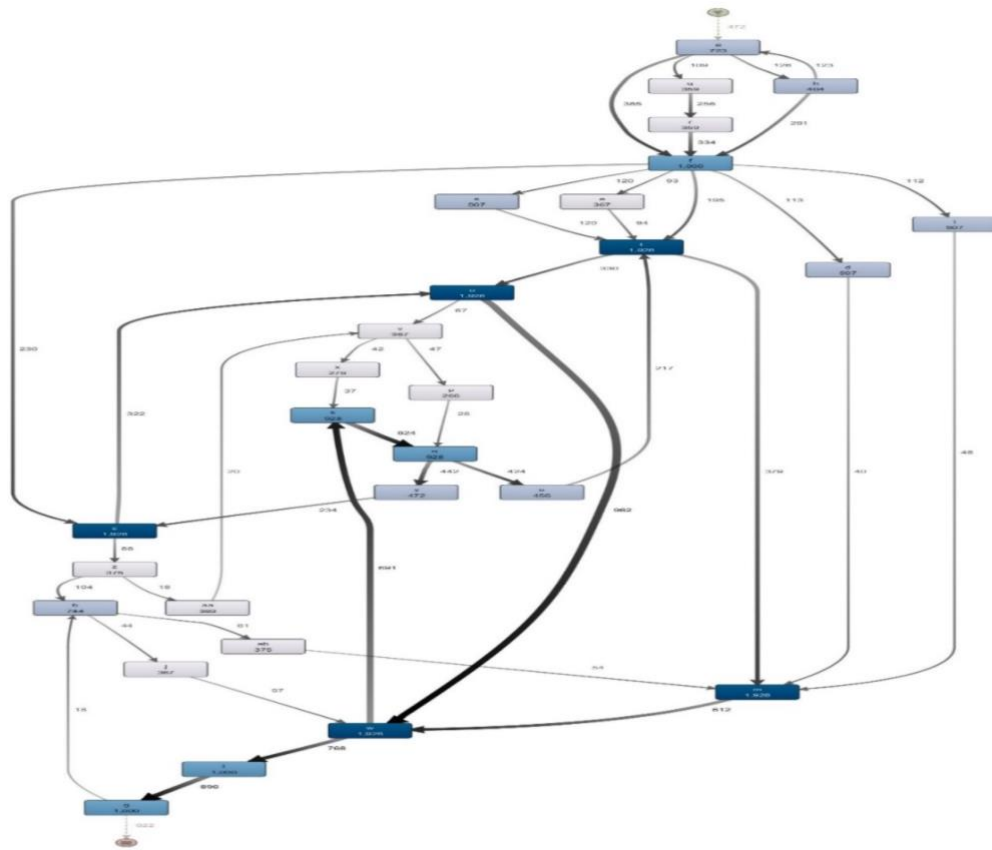


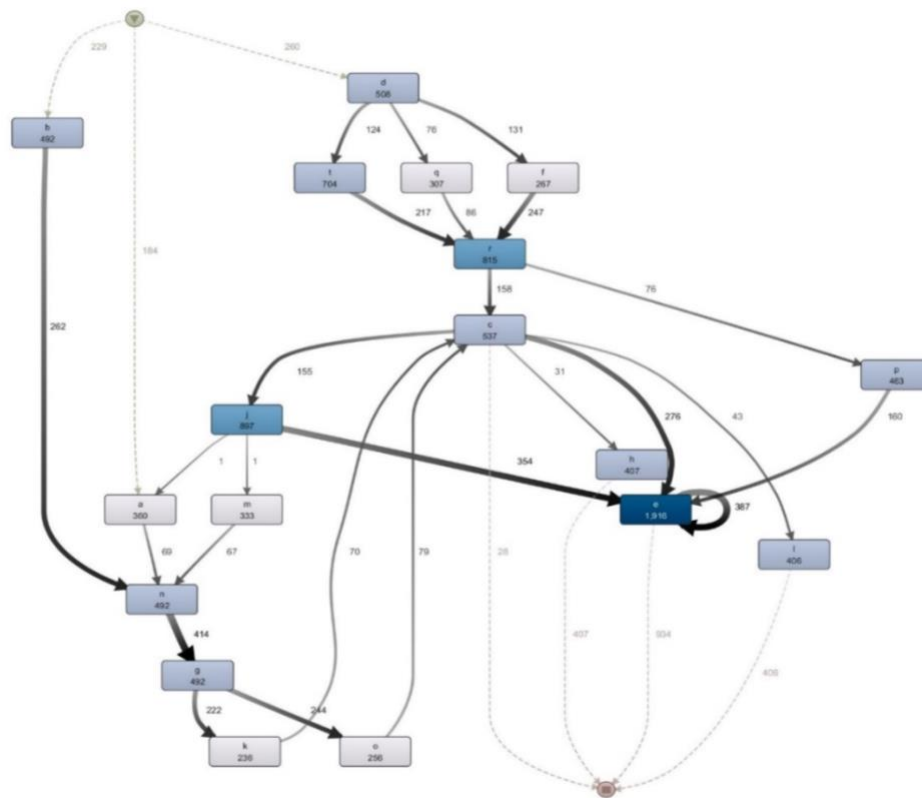
A.2 Fuzzy Model for *training_log_2*

A.3 Fuzzy Model for *training_log_3*

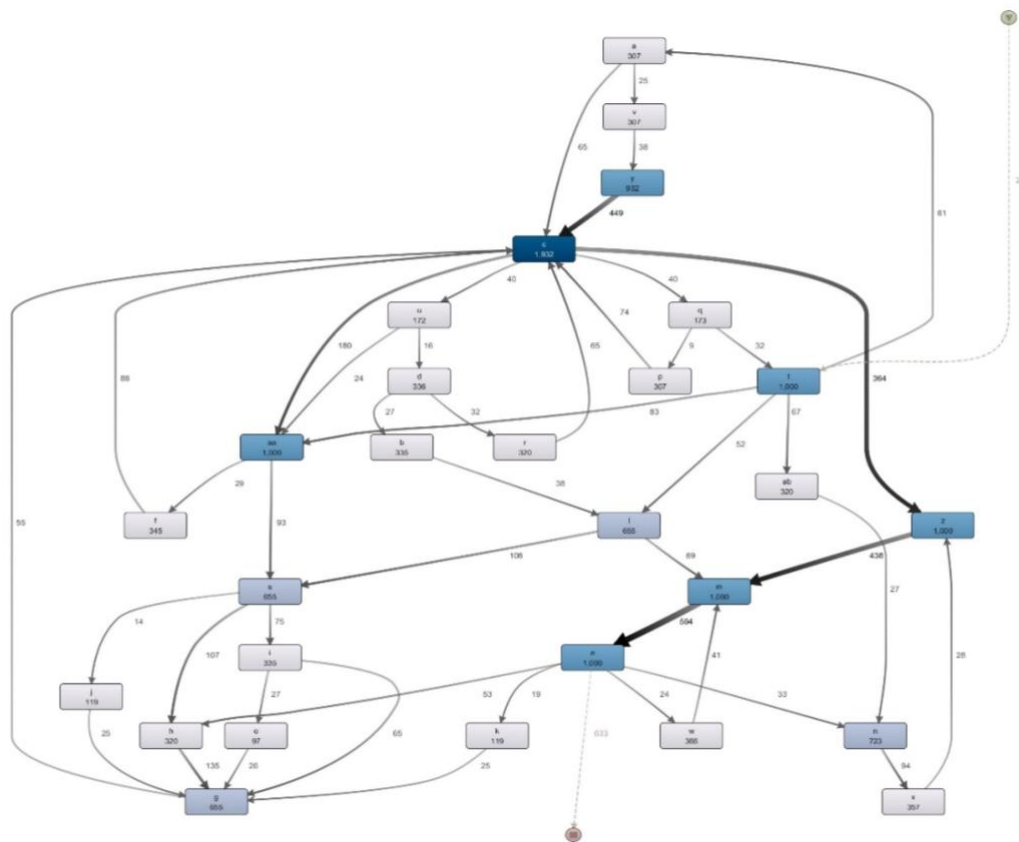


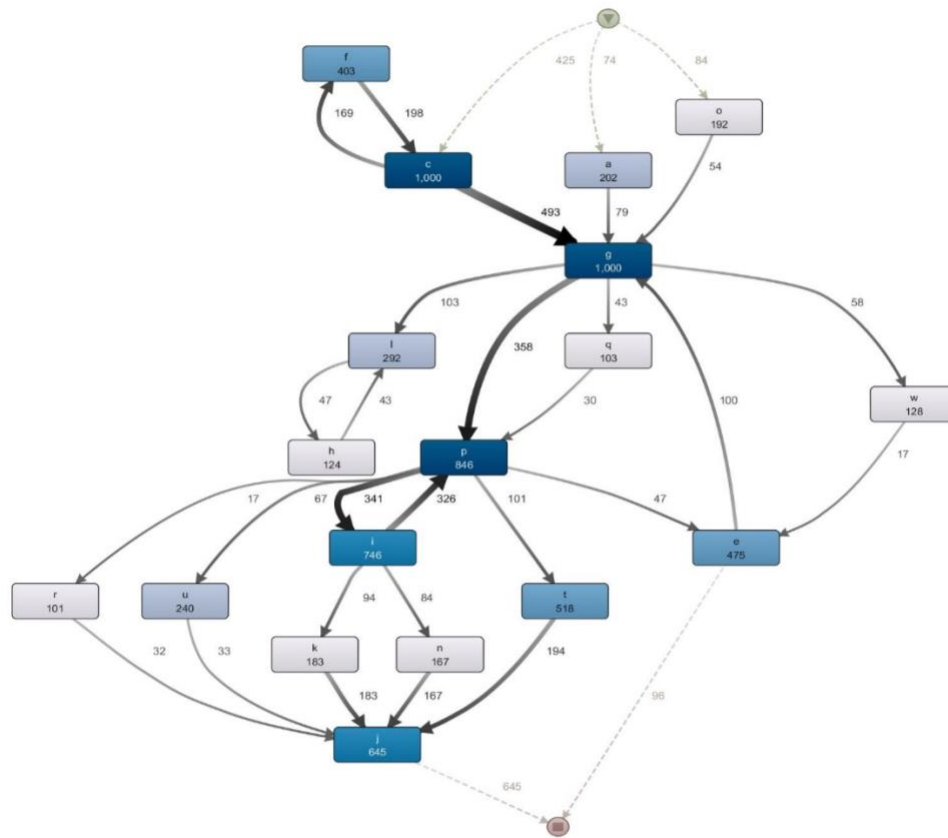
A.4 Fuzzy Model for *training_log_4*

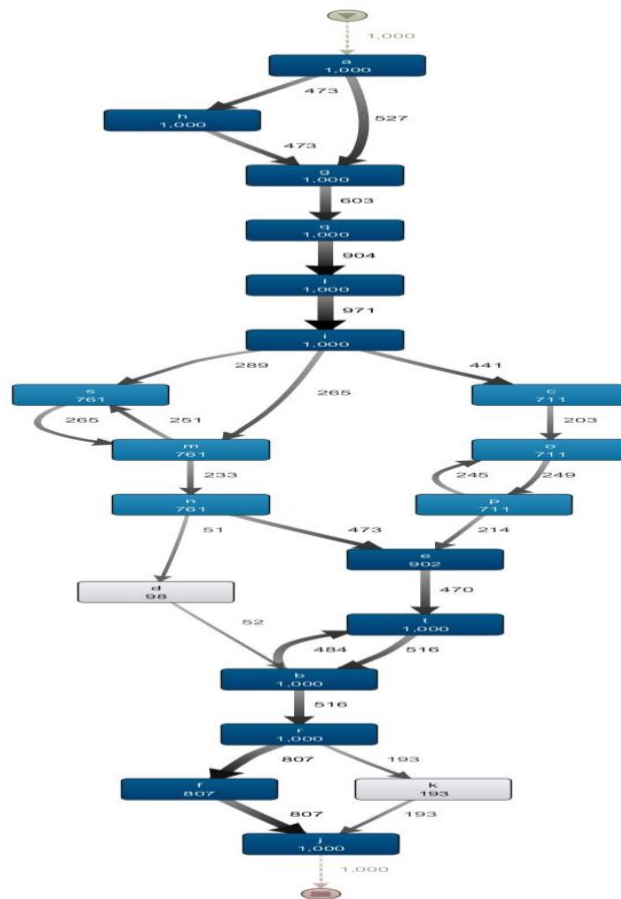
A.5 Fuzzy Model for *training_log_5*

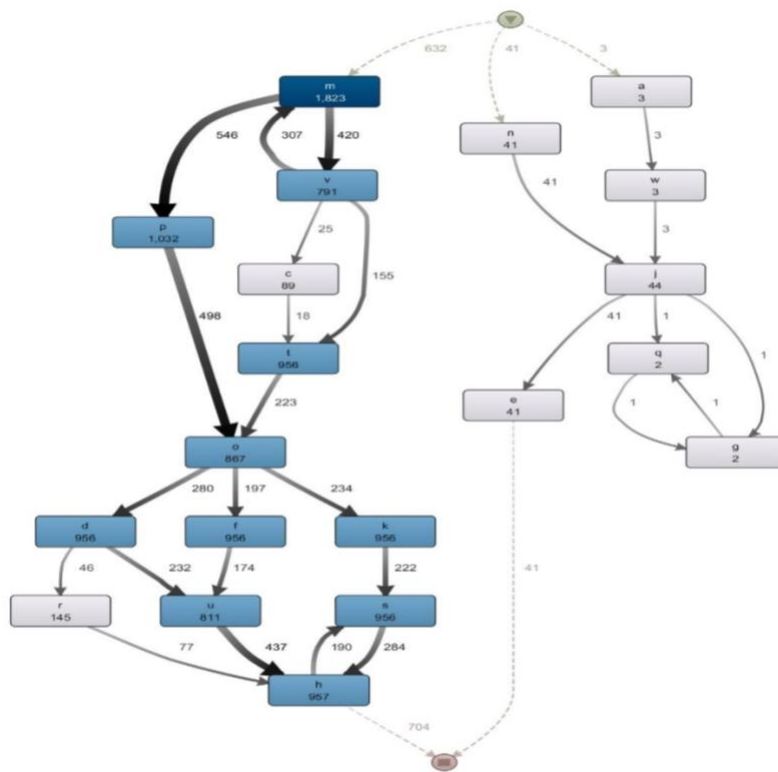


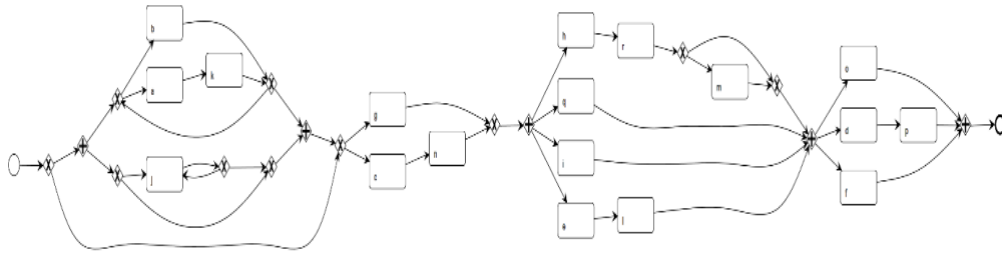
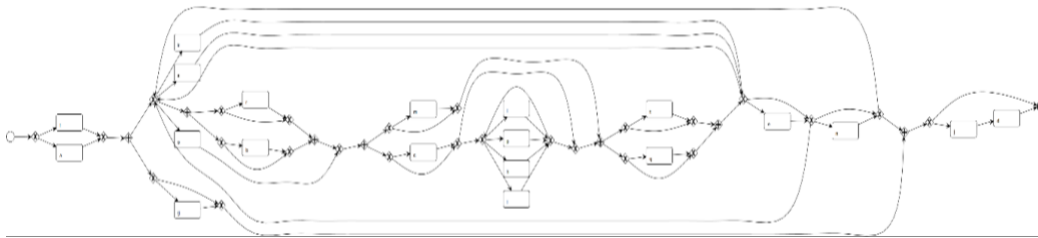
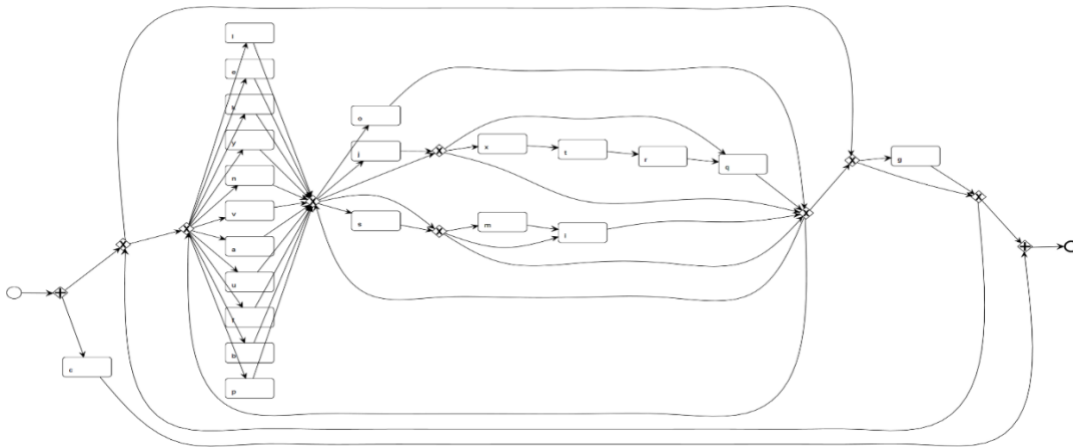
A.6 Fuzzy Model for *training_log_6*

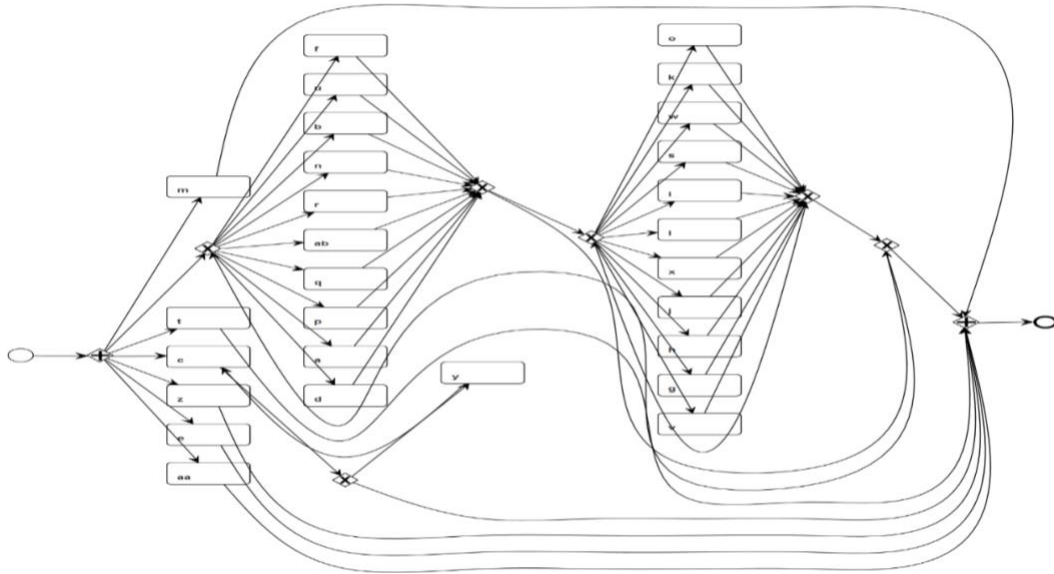
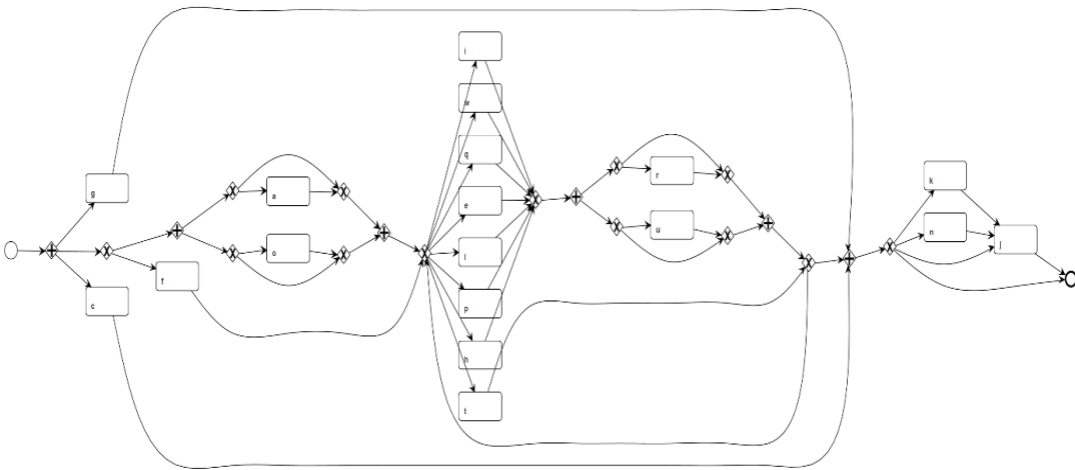
A.7 Fuzzy Model for *training_log_7*

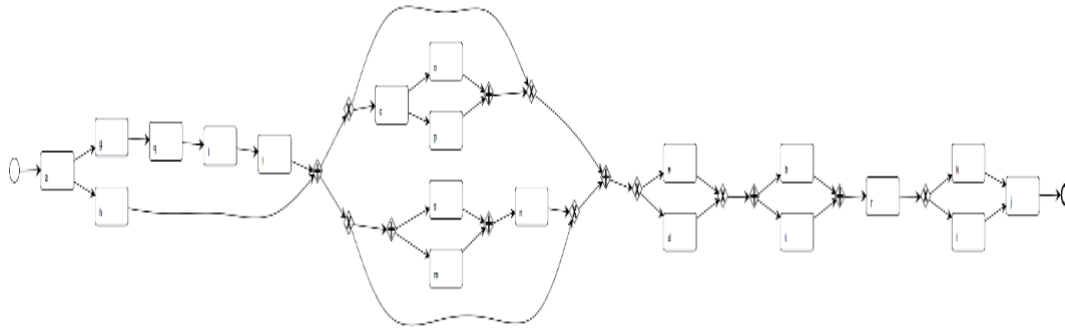
A.8 Fuzzy Model for *training_log_8*

A.9 Fuzzy Model for *training_log_9*

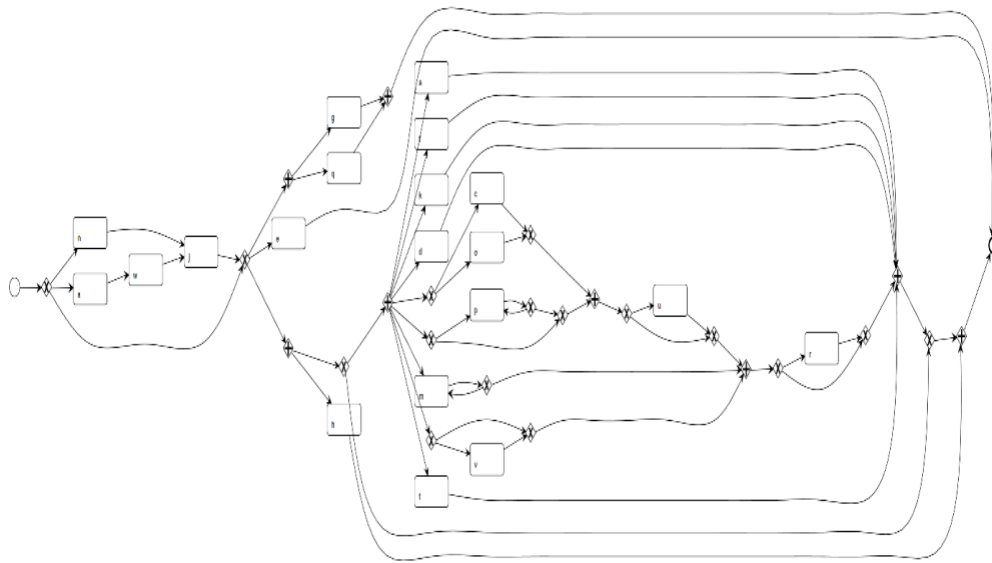


APPENDIX B.**BPMN MODELS FOR THE TRAINING LOGS**B.1 BPMN model for *training_log_1*B.2 BPMN model for *training_log_2*B.3 BPMN model for *training_log_3*

B.7 BPMN model for *training_log_7*B.8 BPMN model for *training_log_8*



B.9 BPMN model for training_log_9



B.10 BPMN model for training_log_10