International Journal of Innovative
Computing, Information and Control
Volume **16**, Number **3**, June **2020**

ICIC International ©2020 ISSN 1349-4198
pp. 899–921

SEMANTIC-BASED PROCESS MINING TECHNIQUE FOR ANNOTATION AND MODELLING OF DOMAIN PROCESSES

Kingsley Okoye[1,2,*], Syed Islam[2], Usman Naeem[3] and Mhd Saeed Sharif[2]

[1]Writing Lab, TecLabs, Office of the Vice President for Research and Technology Transfer
Tecnologico de Monterrey
Monterrey, Nuevo Leon 64849, NL, Mexico
*Corresponding author: kingsley.okoye@tec.mx

[2]School of Architecture Computing and Engineering
College of Arts Technology and Innovation
University of East London
London, E16 2RD, United Kingdom
{ syed.islam; s.sharif }@uel.ac.uk

[3]School of Electronic Engineering and Computer Science
Faculty of Science and Engineering
Queen Mary University of London
Mile End Road, London, E1 4NS, United Kingdom
u.naeem@qmul.ac.uk

Received November 2019; revised March 2020

Abstract. *Semantic technologies aim to represent information or models in formats that are not just machine-readable but also machine-understandable. To this effect, this paper shows how the semantic concepts can be layered on top of the derived models to provide a more contextual analysis of the models through the conceptualization method. Technically, the method involves augmentation of informative value of the resulting models by semantically annotating the process elements with concepts that they represent in real-time settings, and then linking them to an ontology in order to allow for a more abstract analysis of the extracted logs or models. The work illustrates the method using the case study of a learning process domain. Consequently, the results show that a system which is formally encoded with semantic labelling (annotation), semantic representation (ontology) and semantic reasoning (reasoner) has the capacity to lift the process mining and analysis from the syntactic to a more conceptual level.*
Keywords: Semantic annotation, Ontologies, Reasoner, Process mining, Process modelling, Learning process, Event logs

1. **Introduction.** Today, process mining (PM) [1] has become a valuable technique used to discover meaningful information or models from the readily available events log stored in many IT systems. The PM combines techniques from the computational intelligence which has been lately considered to encompass artificial intelligence (AI) or even the latter, augmented intelligence systems, and data mining (DM) to process modelling as well as several other disciplines in order to analyze event data logs. Nonetheless, a common challenge with most of the existing process mining and analysis techniques is that they depend on tags (e.g., labels) in event logs information about the processes they represent, and therefore, to a certain extent are limited because they lack the abstraction level required from real-world perspectives. This means that the techniques do not technically

DOI: 10.24507/ijicic.16.03.899

gain from real knowledge (semantics) that describe the tags in the events log of the domain processes in question.

For this purpose, this paper explores the technological potentials and prospects of using the semantic-based tools or method to manage perspectives of the process mining and the resultant process models – using a case study of the learning process. In essence, the paper addresses the challenges posed by the traditional process mining and analysis techniques by providing a method that considers and focuses on integrating semantic technologies within the existing process knowledge-bases. Thus, the proposed method allows for analysis of the readily available events log about the domain processes based on concepts rather than the tags or labels in the events log of the process. Besides, [1-4] note that an accurate exploration and analysis of the extracted events log is capable of providing useful information with regards to the quality of support being offered for the so-called organizations or the process owners as well as the information systems at large.

Nowadays, a greater number of derived models in many information systems tend to support just machine-readable systems rather than machine-understandable systems at large. Perhaps, 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 trail to inclusively process the information that they contain or support. Moreover, an adequate knowledge-base system is one which is (i) understandable by humans, and (ii) understandable by machines. This means that the process models are either semantically labelled (annotated) to ease the analysis process, or represented in a formal structure (ontology) which allows a computer (e.g., the reasoner) to infer new facts by making use of the underlying relations.

Technically, the method of this paper is realized by defining formats (semantic viewpoints) on the level of systems performance (i.e., domain processes in view) and the sets activities executions in relation to how the processes have been performed (process work flows) [5]. In turn, the semantic modelling process provides us with the opportunity to develop intelligent methods/algorithms that are capable of analyzing the resulting models through explicit specifications of the different process elements otherwise referred to as conceptualization [6-9]. Specifically, [8,9] show the significance of such an ontology-based approach. According to the results [8,9], the ontology-based method involves semantic descriptions and/or reformulation of the meanings of the labels/attributes in the events logs and models as well as their comparisons for the purpose of improving the usefulness and performance of the domain processes in general.

Moreover, there have been some existing gaps in the literature that motivate the work done in this paper, for example, the problems which are associated with information retrieval and extraction from large growing databases [21]. According to [21], a vast number of such systems constructing large knowledge-bases continuously grow, and most often, they do not contain all of the facts for each process element (instance) representation, thereby, resulting in some kind of missing value datasets. In other words, a well-designed information retrieval (mining) system perhaps should present the results and/or discovered patterns in a formal and structured format with the capacity of being interpreted as domain knowledge or to further enhance the existing knowledge-base [2]. Basically, one of the main challenges with the methods which are used to perform information retrieval and extraction is that they rely exclusively on the syntax of labels in the databases, and are very sensitive to data heterogeneity, label name variation and frequent changes [22]. As a result, a number of the resultant process models are discovered without some kind of hierarchy or structuring. Nonetheless, to address the aforementioned problems, [22] links labels in event logs to the underlying semantics that describes the discovered models, in order to provide a more accurate mining and compact analysis of the said processes

at different levels of abstraction. Moreover, [22] proposes a semi-automatic procedure used to associate semantics to training labels through the extraction of process models annotated with semantic information. In the experiment, [22] uses the Ontology Abstract Filter plug-in in ProM [23] as input to a semantically annotated log to produce as output an event log where the names of tasks in the training labels are replaced by the names of a set of chosen concepts. The produced log is then exported as semantically annotated mining extensible markup language (SA-MXML) [3] file format that consequently allows for performing a control-flow mining. In addition, the control-flow mining is done by using the heuristic miner algorithm [24,25] to extract and interpret the process models based on the concepts that have been defined.

Equally, there are problems associated with the methods such as the semantic web search technologies (that trail to combine the information extraction (IE) [36] and information retrieval (IR) [37] methods to find meaningful information or files from large collections of databases, and then present the output/results to the users based on some pre-specified information need). For example, [39] notes that semantic web technologies (e.g., knowledge and information management system (KIM) [39,41], SemTag system [42] and Magpie [43]) are not only useful to add machine tractable or repurposable layer of annotations that are relative to ontologies, but are at the same time required to match or complement the overwhelming (omnipresence) web of natural language hypertext [40]. Perhaps, this is done by creating semantically annotated terms and then linking the resulting pages to ontologies. Moreover, the web ontology language (OWL) [10,27] has emerged as the standard format for defining the semantic web ontologies, and has since in recent years, widely been accepted and particularly utilized towards advanced structuring of information or process engineering/modelling. Indeed, the combined idea of IE and IR is the mechanism upon which the semantic web search methods such as the semantic-based approach proposed in this paper are built.

This paper applies the method (i.e., semantic-based annotation and modelling of the domain processes) on a case study of the learning process domain in order to demonstrate the usefulness of the proposed approach. Essentially, the method takes account of the different stages of the process mining and models analysis (i.e., from the initial phase of collecting and transformation of the readily available events data log to discovering of useful process models) and then carries out semantical annotation of the extracted models for further analysis and querying at a more abstraction level. By the abstraction levels of analysis, the work shows that the semantic-based approach is able to provide an easy and accurate way to analyze the datasets (i.e., the event logs and models) by allowing the meaning of the process elements to be enhanced through the use of property descriptions languages and schema, such as the web ontology language (OWL) [10], semantic web rule language (SWRL) [11], and description logic (DL) queries [12]. This is done in order to make available inference knowledge which is then utilized to determine useful patterns by means of the semantic reasoning aptitudes.

The rest of the paper is structured as follows. In Section 2, the work provides background information and discusses appropriate related works within the field of semantic technologies and its main application areas. Section 3 describes the main method and proposes sets of algorithms used for ample implementation of the semantic-based approach in this paper. In Section 4, the study describes how the semantic-based approach is applied in real-world settings to show the usefulness of the method. Also, the work shows how it utilized the case study of the learning process to illustrate the proposed method. Section 5 discusses the overall results and experimentation outcomes of this paper – particularly by weighing the results of the method against other benchmark algorithms/approaches

used for process mining. Finally, the paper concludes and highlights the limitations to study and directions for future works in Section 6.

2. **Background Information.** The work discusses in this section, the different technologies or methods that have been dedicated to modelling/analysis of the events logs and process models especially as it concerns the semantic-based approaches and process mining techniques.

2.1. **Semantic annotation and data labelling.** One of the biggest challenges when performing process mining and analysis task is to discover the correct information and to comprehend (understand) what they mean [13-15]. According to Rozinat [13], it could be anything between really easy or very complicated to figure out the semantics (metadata) information from existing logs in many IT systems. Besides, the outcomes of such a method often depend on how distant the logs are from the actual data labels or tags (annotation). The works in [16-19] show that the annotated logs or models are necessary for the semantic-based process analysis, and consequently, model enhancement to follow. Specifically, [16] notes that the semantic annotations or yet still, data labelling is an essential component in realizing such methods that support the semantic-based process mining approach by automatically conveying the formal structures of the derived models or extracted logs. In theory, Lautenbacher et al. [17,18] state that semantic annotation is defined formally as a function that returns a set of concepts from the ontology for each node or edge in the resultant graph/models. Whereas Born et al. [19] note that the semantic annotation process could be carried out either manually, or automatically computed bearing in mind the similarity of words to generalize the individual entities within the domain processes in view. Recently, Jonquet et al. [20] have studied ontology metadata practices by analyzing metadata annotations of different ontologies and reviewing the most standard and relevant vocabularies. [20] systematically compares different metadata implementation in various ontology repositories (reference libraries) in order to build a new descriptive model that can be used to describe ontologies.

Likewise, the work in this paper introduces a semantic-based method that transforms the extracted datasets and models into minable executable formats (through the use of property description languages) to support the discovery of improved or enhanced process models. In other words, the proposed technique for annotating the unlabelled activity sequences of this paper uses the ontology schema/vocabularies (e.g., OWL, SWRL, DL queries, restriction properties) to provide metadata or object property assertions that allows for the discovering of useful information or class expressions in existing knowledge-bases through the semantic reasoning aptitudes.

2.2. **Semantic-based process analysis and knowledge engineering.** Indeed, existing works in the literature show that effective methods for semantic-based process mining and analysis should focus on information about resources hidden within the process knowledge-base, and how they are related [3,4,9,21,22]. Typically, the techniques for the semantic-based analysis allow the meaning of the domain entities and object properties to be enhanced through the use of property characteristics and classification of discoverable entities. Essentially, the method is applied in order to permit for analysis of the extracted event logs and models based on concepts rather than the event tags or labels about the process. Currently, there are not too many algorithms that support such semantic analysis and there are few existing applications that demonstrate the capabilities of the semantic-based approach [3,4,8,16,21,23]. The works in [16,23] show how semantic annotations and reasoning can be used to provide a more improved analysis (enhancements) to process models and event logs through concept matching (e.g., ontology population and

classifications). Specifically, the work in [23] shows how to perform the semantic modelling and integration of the resulting process mappings (ontology graphs) with annotated terms and then present the domain knowledge for the activity workflows and concepts defined in the underlying ontology by using process description languages such as the OWL [10] and SWRL [11]. Indeed, reasoning on the ontological knowledge plays an important role in the semantic representation of processes [24]. Besides, semantic reasoning allows for the extraction and conversion of explicit information into some implicit information, for example, the intersection or union of classes, description of relationships and concepts or role assertions.

In short, any ontology-based systems should not only contain the information about the specific domains which they represent but should also provide information about the identified instances (process elements) as well as their individual properties. In other words, an effective ontology-based system must contain a set of well-defined components (e.g., classes) with their full semantic descriptions [9].

3. **Method.** Typically, this work shows that much of the effort in developing a semantic-based process mining and analysis method relies mainly on constructing an effective system that integrates the three main building blocks, namely: annotated logs or models, ontologies and semantic reasoning. In fact, the thematic focus and targeted goal of the method described in this paper and its main application come in well-defined stages as described in Figure 1 and are subsequently illustrated in detail in the next subsection of the paper.

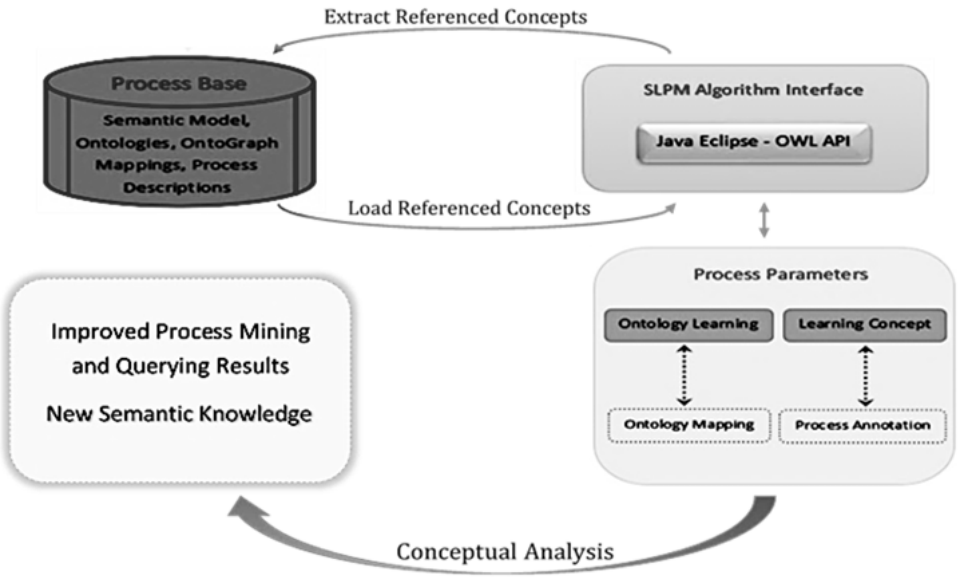

FIGURE 1. Main aspects of implementing the semantic-based annotation and ontological modelling technique

As gathered in Figure 1, the semantic model which forms the basis of analysis in this paper consists of ontologies or class hierarchies (taxonomies/OntoGraphs) which are loaded into the system using the OWL application programming interface (API). Moreover, the underlying process is a recursive process that can be performed as many times (infinite) based on the predefined users' queries or analysis questions.

Furthermore, whilst the proposed semantic-based annotation and ontological modelling process (Figure 1) and sets of semantically motivated algorithms (see Algorithms 1, 2 and 3) are focused on describing the meaning of the process models and the underlying

relations (Algorithm 1), the events log attributes and assertions (Algorithm 2), and the process description and analysis (Algorithm 3) which are all devoted to binding together the different concepts (classes) that make up the defined model. Moreover, the semantic approach focuses on ways that maximize the use, effect, and outcomes of the method particularly from a real-world process point of view by making use of case study of the learning process to demonstrate the method.

In summary, the main components realized as a result of implementing the proposed method as described in Figure 1 are as follows:

(i) Event Logs and Models – show how process mining is applied to improving the informative value of real-time process data.

(ii) Process Mapping (modelling) – describes how improved process models can be derived from the large volumes of event data logs found within the different process domains.

(iii) Semantic Annotation – describes how semantic descriptions (annotation) of the deployed models can help enrich the result of the process mining and outcomes through discovering of new knowledge about the process elements and the underlying relations.

(iv) Ontology – use of ontologies with effective semantic reasoning to lift process mining analysis from the syntactic level to a more conceptual level.

(v) Sets of semantically motivated algorithms – reveal how references to the ontologies can help address the problem of analyzing the events logs or models based on concepts and to answer questions about relationships the process instances share amongst themselves within the knowledge-base.

3.1. **Method for semantic annotation and modelling of process models.** The purpose of semantical annotation of the process models is to provide metadata (process descriptions) that can help represent the events data logs and the discovered models in a formal and structured manner (ontology). The primary aim is to construct a semantic-based reference library (i.e., metadata) for the different process elements (entities), which are then used to support the modelling and analysis of the processes in question. This is done in order to provide domain knowledge (inferences) that can help provide a more conceptual understanding and/or further enrichment of the derived models. Technically, the semantic depiction/representation of the discovered models in an ontological form is a very important step in the method of this paper. In fact, the method is primarily aimed at unlocking the information value of the event data logs (EDL), and the derived process models, M (as described in Algorithm 1) by way of finding useful and previously unknown links between the process elements and the deployed models. Moreover, the use of reasoner to infer the various process instances relies exclusively on the ability to represent such information in a formal way (ontology) in order to create a platform for an enhanced (conceptual) analysis of the individual process elements or knowledge-base. Algorithm 1 describes how the work generates ontologies from the process models and event logs.

From Algorithm 1, we note that ontologies (Ont) are a formal explicit specification of shared conceptualization that can be applied in any context [25]. Indeed, the semantic annotated logs and models are very fitting for further steps of semantically enhancing and carrying out accurate analysis of the process models. This is owing to the fact that at this stage, the input data are represented in a formal and structured format (taxonomies) that can connect to referenced concepts within the ontologies. Interestingly, from the algorithm (Algorithm 1), the work shows that ontologies can be defined as a quadruple, i.e.,

$$\text{Ont} = (\boldsymbol{C}, \boldsymbol{R}, \boldsymbol{I}, \boldsymbol{A})$$

Algorithm 1: Procedure for developing ontologies from process models & event logs

For all defined models $\boldsymbol{M}$ and event log $\boldsymbol{EDL}$
Input: $\boldsymbol{C}$ – different classes for all process domain
$\boldsymbol{R}$ – relations between classes
$\boldsymbol{I}$ – sets of instantiated process individuals
$\boldsymbol{A}$ – sets of axioms which state facts
Output: semantically annotated graphs or semantic model
Procedure: create a semantic model with defined process descriptions and assertions
Begin
  For all process models $\boldsymbol{M}$ and event log $\boldsymbol{EDL}$
    Extract classes $\boldsymbol{C} \leftarrow$ from $\boldsymbol{M}$ and $\boldsymbol{EDL}$
    while no more process element is left **do**
    Analyze classes $\boldsymbol{C}$ to obtain process instances, class hierarchies, and relations
      If $\boldsymbol{C} \leftarrow$ Null **then**
        obtain the occurring process instances ($\boldsymbol{I}$) from $\boldsymbol{M}$ and $\boldsymbol{EDL}$
      Else If $\boldsymbol{C} \leftarrow 1$ **then**
        create the relations ($\boldsymbol{R}$) between objects and data types // i.e., between classes $\boldsymbol{C}$ and individuals ($\boldsymbol{I}$)
      If relations $\boldsymbol{R}$ exist **then**
        For each class $\boldsymbol{C} \leftarrow$ semantically classify or populate extracted relationships ($\boldsymbol{R}$) to state facts, i.e., Axioms ($\boldsymbol{A}$)
        create the class structures by adding the extracted relationships and individuals to the ontology
Return: taxonomy
End If statements
End while
End For

which consists of different classes, $\boldsymbol{C}$, and relations, $\boldsymbol{R}$, between the classes [17,25,26]. Perhaps, a relation, $\boldsymbol{R}$, trails to connect a set of class(es) with either another class (or fixed-literal) and is capable of also describing the sub-assumption hierarchies (taxonomy) that exists between the various classes. In essence, the class(es) are instantiated with a set(s) of individuals, $\boldsymbol{I}$, and can likewise contain a set(s) of axioms, $\boldsymbol{A}$, which states facts (e.g., what is true and fitting within the model, or what is true and not fitting within the model). In other words, ontologies can be defined as connected sets of taxonomies (RDF + Axioms), or yet, structuring in a formal way (Triple + Facts) where the subject includes the defined classes and predicate representation of the relations, whereas the object includes the individuals (process instances) and sets of axioms which states facts. Petrenko and Petrenko [26] are even more specific about the importance of the ontological concepts (particularly classes) in semantic representation of process models. [26] notes that "classes are the central item of the ontology" and further states that a well-defined class may represent all types of procedures, e.g., running tasks, data transmission, data flow control, and activity workflows. In theory, the ontological concepts, process descriptions, relations, etc. as shown in Algorithm 1 show that semantic annotation is an essential way of realizing the ontology-based system (that supports semantic-based process modelling and analysis) by automatically conveying the formal semantics of the derived models and/or extracted logs [17,18]. In other words, the annotated logs or models are necessary for concrete implementation of the semantic-based process mining and analysis [8,9,16]. In principle, semantic annotation of the process models is defined formally as [17]:

$$\mathrm{SemAn} :: \mathrm{N} \cup \mathrm{E} \rightarrow \mathrm{COnt}$$

where SemAn describes all kinds of annotations which can be input, output, meta-model annotation, etc. Besides, semantic annotations can be carried out in different ways (either manually, semi-automatically or automatically) [19] depending on the domain process in question, or the authoring tool/method the process analyst/developers choose. In fact, semantically planning of any ontology-based system such as the method of this paper requires that all process modelling, and definition of the ontologies must include some form of semantic annotation. Moreover, by taking account of the definition in [17] if we let A be the set of all process actions. A process action $a \in A$ is characterized by a set of input parameters $Ina \in P$, which is required for the execution of a and a set of output parameters $Outa \subseteq P$, which is provided by a after execution. All elements $a \in A$ are stored as a triple $(namea, Ina, Outa)$ in a process library $libA$.

To do this, at first, the extracted logs/models from the standard process mining techniques are represented as a set of annotated terms which links or relates to defined terms within an ontology as illustrated in detail in the following figure (Figure 2). Indeed, the method makes it straightforward to represent the extracted information in an easy and yet accurate manner.

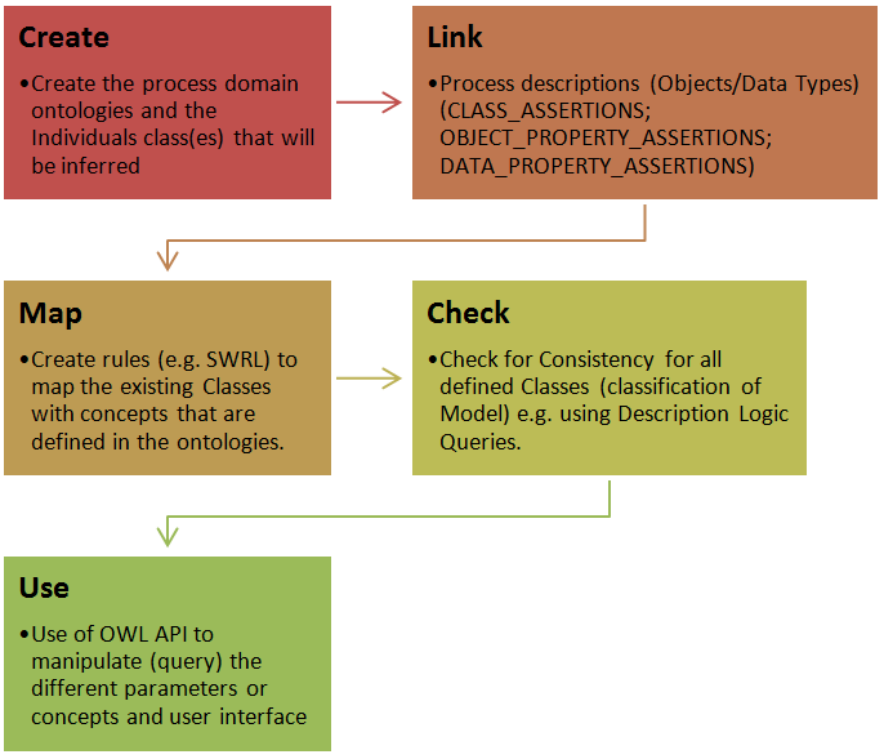

FIGURE 2. Incremental procedure used for implementing the process described in Algorithm 1

Secondly, the resulting ontologies provide means to represent the annotated terms or process in a formal and structured way by defining the associations (relationships) between the different process elements as observed in the model. Perhaps, the method also ensures that the various range of tasks (activities) conforms naturally to the event logs as well as the model representations. This is achieved by encoding the deployed models in a formal structure of ontologies (i.e., semantic modelling), and even more, supports further expansion (or improvement) of the existing model.

Finally, the reasoner (the inference engine) is designed to perform semantic reasoning and ontology classification of the different process elements in order to validate the resulting model and clean out inconsistent outputs, and consequently, presents the inferred (underlying) semantic associations in a structured manner.

Moreover, the work highlights in Figure 2 the incremental contributions and/or main functions of Algorithm 1.

In theory, as gathered in the aforementioned process, the work shows that the first step towards achieving semantic annotation of the derived models should aim at making use of process description languages/assertions (e.g., OWL, SWRL, DL Queries) [10-12] to link elements in the models with concepts that they represent in well-defined ontologies. Practically, the purpose of the method (semantic annotation) is to seek equivalence between the concepts of the derived models with concepts of the defined ontologies.

The following Figure 3 is an exemplary model for the learning process (research domain) which the work uses to illustrate the practical implementation of the method and experimental setup. The process as represented in Figure 3 has been modelled using the business process modelling notation (BPMN). The resulting output (semantic-based annotation of the models) is an ontological model (OntoGraph) which consists of semantic assertions (labelled concepts) as shown in Figure 4. Consequently, in Figure 4 (which is described in detail in Section 4.1), the work makes use of the process description languages such as OWL and SWRL to create the semantic model that represents the model shown in Figure 3.

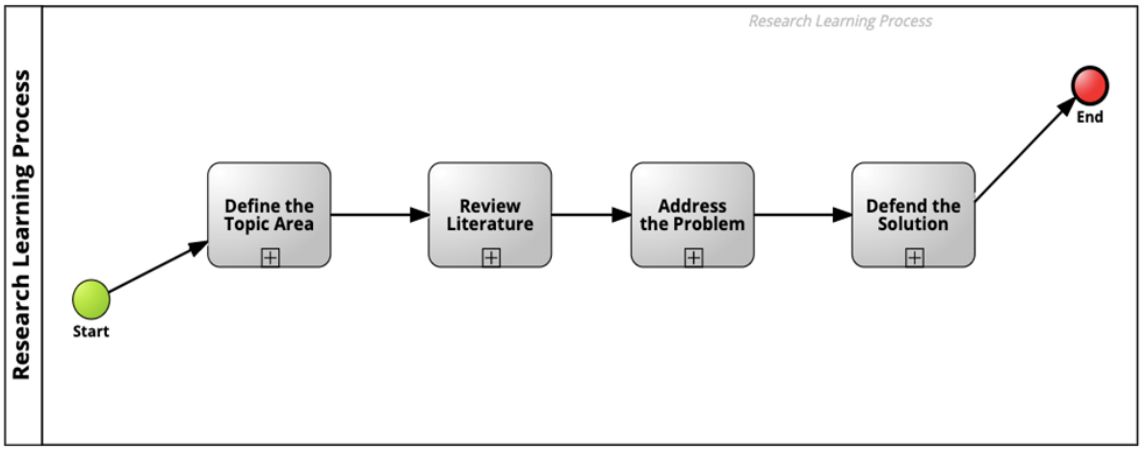

FIGURE 3. Research learning process (input process model)

3.2. **Automated generation of process instances, and class concepts.** Accordingly, Algorithm 2 describes how the work determines the correlation between concepts of the process models and concepts of the defined domain ontologies. This is done using the case study of the learning process. Theoretically, the work shows in the algorithm (Algorithm 2) how by constructing semantic-based models (i.e., with a description of the process elements and concepts), it becomes easy to accurately determine the different patterns or behaviours that can be found within the learning knowledge-base. Moreover, the semantic-based learning process mining (SLPM) algorithm (Algorithm 3) explains the basis for the semantical modelling, integration, and analysis of the different concepts. Although the works have used the case study of the learning process to demonstrate the implementation of the approach, the proposed steps can be applied to any given process domain provided the variables as described in the algorithms (Algorithms 2 and 3) are present in the readily available events logs or models.

Algorithm 2: Generating process instances, classes, and learning activity concepts (AC)

For all definite classes and process descriptions
Input: $\boldsymbol{AC}$, learners prior activity list $\boldsymbol{ACL}$_List
Output: $\boldsymbol{AC}$'s learning activity sequence set $\boldsymbol{LS}$
Procedure: Generate learning activity classes and subsets
Begin
$\boldsymbol{LS}$ = Null
$\boldsymbol{AC}$_ProcessInstance_List = Null
$\boldsymbol{AC}$_LearningActivity = 0
    For all LearningActivity_$\boldsymbol{AC}$_within the knowledge-base
        Extract $\boldsymbol{LS} \leftarrow \boldsymbol{LS} + \boldsymbol{AC}$
    while no more $\boldsymbol{AC}$ is left **do**
        For each $\boldsymbol{Ci} \in \boldsymbol{LS}$
            $\boldsymbol{Ci}$_Precondition_List ← Get_Precondition ($\boldsymbol{OWL_xml_Ci}$)
        For each $\boldsymbol{Cj} \in \boldsymbol{Ci}$_Precondition_List
            $\boldsymbol{Cj}$_CorrespondingSubSet_List = Null
            $\boldsymbol{Cj}$_ProcessInstance_List = Null
            If $\boldsymbol{Cj} \notin \boldsymbol{ACL}$_List **AND** $\boldsymbol{Cj} \notin \boldsymbol{LS}$ **then**
                $\boldsymbol{LS} \leftarrow \boldsymbol{LS} + \boldsymbol{Cj}$
                $\boldsymbol{Cj}$_CorrespondingSubclassSet_List ← $\boldsymbol{Cj}$_CorrespondingSubclassSet_List + $\boldsymbol{Ci}$
                $\boldsymbol{Cj}$_ProcessInstance_List ← $\boldsymbol{Cj}$_ProcessInstance_List + $\boldsymbol{Ci}$ + $\boldsymbol{Ci}$_ProcessInstance_List
                $\boldsymbol{Cj}$_LearningActivity = $\boldsymbol{Ci}$_LearningActivity + 1
            Else If $\boldsymbol{Cj} \notin \boldsymbol{ACL}$_List **AND** $\boldsymbol{Cj} \notin \boldsymbol{LS}$ **AND** $\boldsymbol{Cj} \notin \boldsymbol{Ci}$_ProcessInstance_List **then**
                $\boldsymbol{Cj}$_CorrespondingSubclassSet_List ← $\boldsymbol{Cj}$_CorrespondingSubclassSet_List + $\boldsymbol{Ci}$
                $\boldsymbol{Cj}$_ProcessInstance_List ← $\boldsymbol{Cj}$_ProcessInstance_List + $\boldsymbol{Ci}$ + $\boldsymbol{Ci}$_ProcessInstance_List
                If $\boldsymbol{Cj}$_LearningActivity < $\boldsymbol{Ci}$_LearningActivity + 1
                    For each $\boldsymbol{Ck} \in \boldsymbol{LS}$_*SubsequentTo*_$\boldsymbol{Cj}$
                $\boldsymbol{Ck}$_LearningActivity = **All** ($\boldsymbol{Ck}$_CorrespondingSubclassSet_LearningActivity) + 1
Return $\boldsymbol{LS}$
End For
End If
End For
End For
End while
End For

In principle, the work refers to the learning process as a workflow (sequence of steps) or set(s) of activities through which the learners have to perform in order to complete the research process [27]. To this effect, it was necessary to provide pre-defined activity concepts, $\boldsymbol{Ci}$, (e.g., classes) to be able to identify and/or monitor the entire process flow, and in turn, help in classification of the sets of individual entities (process instances) that make up the defined class.

Therefore, the learning activity concepts and class generation method (Algorithm 2) outline the procedures that take place when generating the lists of process instances and/or defined concepts, $\boldsymbol{Ci}$, within the learning knowledge-base. Henceforth, for each concept

(or class) $\boldsymbol{Ci}$, within the knowledge-base, we first extract the preconditional (prerequisite) list from its OWL file descriptions $\boldsymbol{OWL\_xml\_Ci}$ as shown in line 13. Then for each concept $\boldsymbol{Cj}$ within the class list (e.g., the individual process instances) if it does not belong to an activity list and the corresponding subclass sets, add it into the learning activity sets and revise the $\boldsymbol{Cj}$'s corresponding SubclassSet list, process instance list, and number of steps to the targeted learning concepts as described in lines 14 to 21. Moreover, if $\boldsymbol{Cj}$ already exists in the learning class list, but does not belong to the activity list and individual (process instance) list of $\boldsymbol{Ci}$, end the process, but also update its corresponding subclasses, process instance list, and number of steps to the target learning concepts as described in lines 22 to 27 (Algorithm 2).

Furthermore, if we use the following standard notation, $\boldsymbol{R}$, to refer to the research learning process, and $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$ for the activity concepts (see Algorithm 3). Then $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d} \in \boldsymbol{R}$ is a function with domain $\boldsymbol{R}$ and learning process milestones or attributes $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$, where:

Domain $\boldsymbol{R}$ is a SuperClass of the SubClasses $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$ as described in line 2.

Algorithm 3: Semantic-based learning process mining (SLPM) and analysis procedure

1: For all defined classes or subsets
2: **Input:** $\boldsymbol{\mathcal{L}}$ – process log for *Person*, $\boldsymbol{P}$, over *Researchprocess*, $\boldsymbol{R}$
$\boldsymbol{a}$ – DefineTopicArea Milestone or SubClass
$\boldsymbol{b}$ – ReviewLiterature Milestone or SubClass
$\boldsymbol{c}$ – AddressProblem Milestone or SubClass
$\boldsymbol{d}$ – DefendSolution Milestone or SubClass
3: **Output:** Structured (superClass -> subClass hierarchies) representation and analysis of the research process.
4: **Procedure**: create activities sets that make up research process, $\boldsymbol{R}$
5: **Begin**
6: **For all** Learning Activity concepts $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d} \in \boldsymbol{R}$
7: **If** $\boldsymbol{P} \ldots \boldsymbol{n}$ is a measure of the number of times $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}$ occurs in $\boldsymbol{R}$ for *Person*, $\boldsymbol{P}$, **then**
$\boldsymbol{P} \ldots \boldsymbol{n} = |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R}|$ where, $\boldsymbol{P} \ldots \boldsymbol{n} = |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}a}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}b}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}c}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}d}|$
8: **while** no more process element is left **do**
9: **Run** Reasoner to infer classes and obtain formal structures
10: **If** $\boldsymbol{PSL} = |SL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R}|$ where, $\boldsymbol{PSL} = |SL \subseteq \boldsymbol{\mathcal{L}a}| + |SL \subseteq \boldsymbol{\mathcal{L}b}| + |SL \subseteq \boldsymbol{\mathcal{L}c}| + |SL \subseteq \boldsymbol{\mathcal{L}d}|$ **then**
11: *Person* $\boldsymbol{P}$, is SuccessfulLearner
12: **Else If** $\boldsymbol{PUL} = |UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - 1|$ where, $\boldsymbol{PUL} = |UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{a}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{b}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{c}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{d}|$ **then**
13: *Person* $\boldsymbol{P}$, is UnCompleteLearner
14: **For** each learner class or subsets
15: update the class hierarchies (taxonomy) by adding the extracted relationships/ individuals to the ontologies
16: **Return**: taxonomy
17: **End If** statements
18: **End while**
19: **End For**

Perhaps, we note that the Subclasses (also referred to as subsets) is a set where each of the individual learning activity occurs and sometimes may occur multiple times.

For example, the following activities $\boldsymbol{a1}$, $\boldsymbol{a2}$, $\boldsymbol{a3}$, $\boldsymbol{a4}$, $\boldsymbol{a2}$, $\boldsymbol{a5}$ could be seen as a sequence set of learning activities for *Person*, $\boldsymbol{P} \ldots \boldsymbol{n}$ over $\boldsymbol{a}$ (the DefineTopicArea Milestone). Thus,

$$\boldsymbol{P} \ldots \boldsymbol{n}(\boldsymbol{a}) = |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}a}| \text{ (Line 7).}$$

TABLE 1. Class expressions (process descriptions) for the process instances

Learner Category Class		
	Necessary condition	**Necessary and sufficient condition**
SuccesfulLearners class	Every SuccessfulLearner is a LearnerCategory Every SuccessfulLearner isPerformerOfs an ActivityConcept	Every SuccessfulLearner is something that hasCompleteMilestone an AddressProblem And that hasCompleteMilestone a DefendSolution And that hasCompleteMilestone a DefineTopicArea And that hasCompleteMilestone a ReviewLiterature
	Necessary condition	**Necessary and sufficient condition**
UncompleteLearners class	Every UncompleteLearner is a LearnerCategory Every UncompleteLearner isPerformerOfs an ActivityConcept	Every UncompleteLearner is something that hasOnlyCompleteMilestone an AddressProblem Or that hasOnlyCompleteMilestone a DefineTopicArea Or that hasOnlyCompleteMilestone a ReviewLiterature

So therefore, if $\boldsymbol{a1}$ = Define Topic, $\boldsymbol{a2}$ = Approval Activity, $\boldsymbol{a3}$ = Topic Decline, $\boldsymbol{a4}$ = Refine Topic, $\boldsymbol{a5}$ = End Topic Proposal. *Then*, the sequence set of activities for $\boldsymbol{P}\ldots\boldsymbol{n}(\boldsymbol{a})$ = {Define Topic, Approval Activity, Topic Decline, Refine Topic, Approval Activity, End Topic Proposal}.

On the one hand, the learning problem which this work trails to resolve is on computing the set(s) of individual process instances that have completed (successful learners) or not completed (uncomplete leaners) the learning process, $\boldsymbol{R}$. Essentially, as described in line 7 of the algorithm (Algorithm 3), the work notes that to complete a process, $\boldsymbol{R}$ (i.e., the superClass) one must complete a set(s) of given milestones (i.e., the subClasses $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$) and must perform the set (or perhaps a subset) of the activities that comprise it. Thus, the sum or difference in process logs or activities for any named *Person*, $\boldsymbol{P}$, is defined as follows:

$$\boldsymbol{P}\ldots\boldsymbol{n} = |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}a}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}b}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}c}| \pm |\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}d}|.$$

Thus, $\boldsymbol{P}\ldots\boldsymbol{n}$ is a finite set $|\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R}|$. (Line 7)

For instance, the work defines in line 10 of Algorithm 3 and as explicated in Table 1 that "Every Person that hasCompleteMilestone a DefineTopicArea and that hasCompleteMilestone a ReviewLiterature and that hasCompleteMilestone an AddressProblem and that hasCompleteMilestone a DefendSolution is a SuccessfulLearner".

Hence, for any individual to become a member of the Class SuccessfulLearners, $\boldsymbol{PSL}$, the sum of a set of activities log, $\boldsymbol{\mathcal{L}}$, that the learner has completed must be equal to $\boldsymbol{a}$, and $\boldsymbol{b}$, and $\boldsymbol{c}$, and $\boldsymbol{d}$. Thus,

If $\boldsymbol{PSL}$ is the Class that consist of the set $|SL \subseteq \boldsymbol{\mathcal{L}a}| + |SL \subseteq \boldsymbol{\mathcal{L}b}| + |SL \subseteq \boldsymbol{\mathcal{L}c}| + |SL \subseteq \boldsymbol{\mathcal{L}d}|$.

Then $\boldsymbol{PSL}$ is the set $|SL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R}|$. (Lines 10 and 11).

Likewise, the work establishes in lines 12 and 13 and Table 1 that "Every Person that hasOnlyCompleteMilestone a DefineTopicArea or that hasOnlyCompleteMilestone a

ReviewLiterature or that hasOnlyCompleteMilestone an AddressProblem is an UncompleteLearner".

Therefore, UncompleteLearners, $\boldsymbol{PUL}$, is the class of learners whereby some set(s) of activities for the milestones $\boldsymbol{a}$, or $\boldsymbol{b}$, or $\boldsymbol{c}$, or $\boldsymbol{d}$ is missing over a finite set $|\boldsymbol{n} \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R}|$. Thus,

If $\boldsymbol{PUL}$ is a Class that consist of the set $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{a}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{b}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{c}|$ or $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - \boldsymbol{d}|$,

Then $\boldsymbol{PUL}$ is the set $|UL \subseteq \boldsymbol{\mathcal{L}} \in \boldsymbol{R} - 1|$. (Lines 12 and 13).

Nonetheless, Table 1 is the class expressions (i.e., properties description or assertions) for the Successful and Uncomplete Learners class as defined within the resultant model used for the experimentations in this paper. The table (Table 1) shows the different attributes for the defined category of learners or concepts as a result of applying the semantic-based method.

4. **Experimental Setup and Case Study Implementation.** In this paper, the case study example of the learning process is used to show the usefulness of the proposed method and the sets of algorithms formalization. For instance, the method is implemented to determine what attributes or paths the learners (e.g., process instances) follow or have in common, or what attributes distinguish the successful learners from the uncomplete ones (as described in Table 1). The purpose is not only to answer the specified learning questions (e.g., as defined in Algorithm 3 and Table 1) by using the semantic-based approach, but to show how by referring to the attributes or concepts (Algorithms 1 and 2) it becomes easy to refer to a particular case (e.g., the learners categories). Interestingly, the procedures described in Algorithms 1, 2 and 3 can be applied to any given process domain or model, as long as there is some form of available events logs and derived models from the processes in question.

4.1. **Semantic modelling and representation of the learning process.** To demonstrate the method for semantic annotation of process models as described in this paper (Section 3), the work defines the four milestones: Establish Context → Learning Stage → Assessment Stage → Validation of Learning Outcome (see Figure 3) in order to explain the steps taken during the research process [4,8,27]. Technically, those milestones consist of a sequence(s) of learning activities, and the order in which the individual activities are carried out has the capability of determining the research outcome. Consequently, Figure 4 represents the Class diagram (taxonomy) for the different milestones (subclasses) of the research process with semantic descriptions (annotations) of the different activity concepts mappings (OntoGraph) and relationship (links) between the process instances. The drive for the semantic mapping of the learning activity concepts is that the method allows the meaning of the learning objects (properties) to be enhanced through the use of property descriptions to populate the ontologies (taxonomy) and classification of the discoverable entities.

For example, as highlighted in Figure 4, the following is the metadata description (object or dataType property) of the DefineTopicArea (Class) concepts and assertions within the research process domain ontology.

As described in the given example (DefineTopicArea Class), the work shows that for any individual entity to complete the milestone or sub-process (DefineTopicArea) it must have some set of descriptive properties (or activities) such as StartResearchProcess, DefineTopic, ApproveResearchProposal, or NotifiyProposalAmmendement, RefineTopicProposal, and then EndDefineTopic, etc.

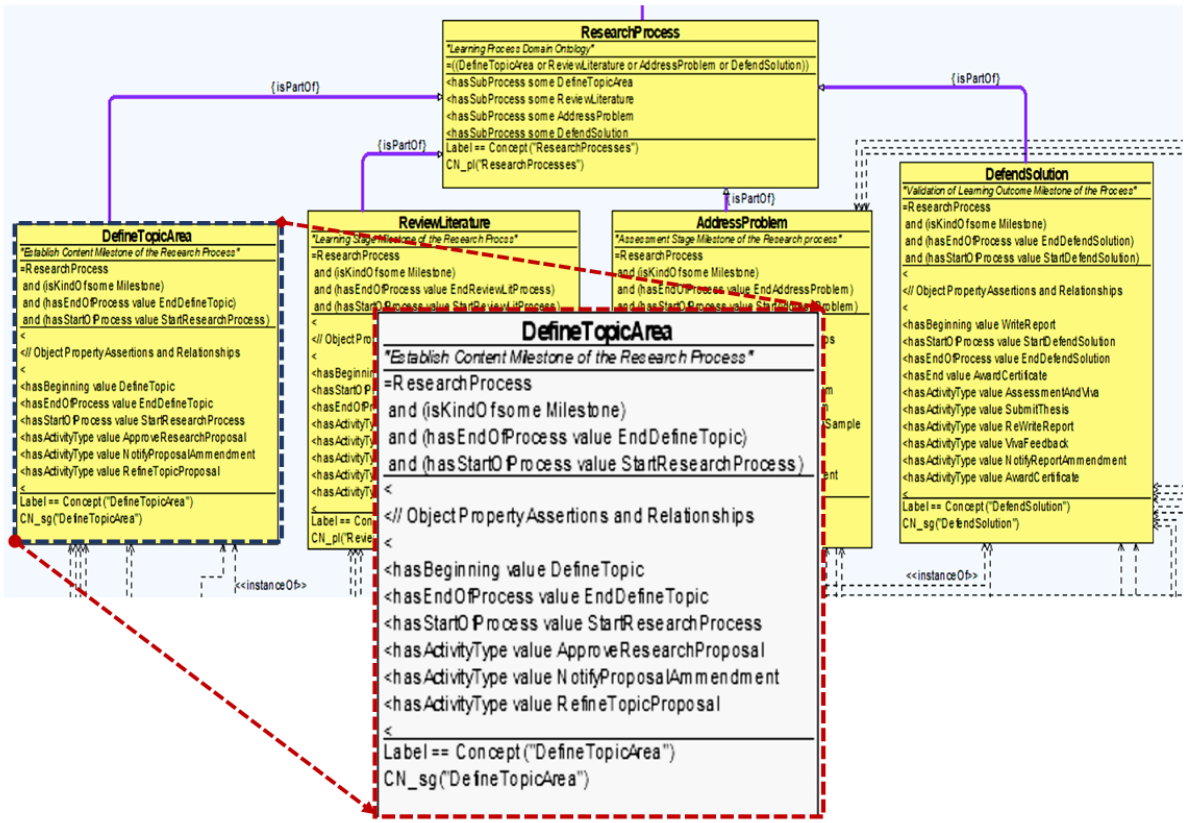

FIGURE 4. Class diagram (taxonomies) for research process domain with description of the concepts and assertions

DefineTopicArea Class
1: **ontology** ResearchProcess
2: **concept** DefineTopicArea
3: **metadata (process descriptions):**
hasBeginning **someValue** DefineTopic
hasEndOfProcess **someValue** EndDefineTopic
hasStartOfProcess **someValue** StartResearchProcess
hasActivityType **someValue** ApproveResearchProposal
hasActivityType **someValue** NotifyProposalAmmenment
hasActivityType **someValue** RefineTopicProposal
4: **axiom**: DefinitionOfDefineTopicAreaMilestone

In fact, regardless of the domain in view or process that one may be analyzing (e.g., case study of the learning process in this paper) we show how the various entities and/or ontological classifications (taxonomies) for any given process are effectively designed, semantically modelled, and developed.

In short definition, property restrictions otherwise referred to as semantic labelling of data or models structuring, stand as a good practice for representation of any given process. This is done by providing a formal way of determining the individual process instances and the relationship they share within the knowledge-base. Moreover, the method can be applied to any given domain as shown in this paper using the case study of the

learning process. On the one hand, not only does this kind of ontology-based representation (formal structuring or taxonomies) support the application of rules such as the SWRL [11] and DL queries [12] and/or re-use of an ontology by another ontology, but on the other hand, it minimalizes the level of human-errors which are every now and again present especially when managing the manifold existence of entities (concepts) within domain processes.

5. **Results and Discussion.** The use of semantic labelling (annotations) and ontologies (Ont $\in$ Onts) including the defined relations ($\boldsymbol{R}$) between concepts ($COnts$) and inferred axioms ($\boldsymbol{A}$) for process modelling have proved beneficial to aggregate tasks and compute the structure (formal) of the derived models and their analysis particularly at the abstraction levels [8,9,25]. The main discussion pertinent to the work in this paper is that for the semantic-based process mining and analysis method, those aspects of aggregating the task [28] or computing the hierarchy of the process models [29,30] should not only be designed to be machine-readable. However, also the methods must, on the other hand, focus on providing a system that is equally machine-understandable [31,32]. In essence, through the semantically annotated logs or use of the process description languages, the method helps to provide metadata as well as integration of the underlying ontologies. Moreover, not only does ontologies allow the creation of the annotations and shared understanding of the domain (process), but it also provides conceptual knowledge that is used to automatize the task within the knowledge-base. Besides, if machines (e.g., computers) can understand the contents of the processes which they support (machine-understandable) then they can also perform more meaningful and intelligent queries and/or analysis.

In principle, the purpose of the semantic annotation as presented in this paper is to seek the equivalence between the different concepts that can be found within the process base or models and concepts of the defined domain ontologies. Moreover, ontology is one, if not only widely accepted tool currently in the literature that is capable of enabling the modelling of uncertainty and imprecisions [33] that often characterizes the human representations of knowledge [25]. Perhaps, we note that by semantically integrating [32,34-36] the process knowledge-bases or datasets with concepts within a well-defined (semantic) model, the resulting systems can make decisions like humans do [32]. For instance, the learning questions addressed in this paper that allows one to determine which entities within the learning process model are classified as successful learners or not. Interestingly, such a method often allied to the ontology-based information extraction systems (OBIE) [2,37] proves to offer solutions that bear the characteristics of "intelligence" which are in many settings usually attributed to humans only. Besides, those characteristics have been considered broadly as a specific feature of *computational intelligence* rather than just an area of the artificial intelligence (AI).

On the one hand, this paper has utilized the ontology-based method to represent information about the different models in a formal structure by making use of the essential building blocks, namely: semantic labelling (annotation), semantic representation (ontology), and semantic reasoning (reasoner). Therefore, the study provides the semantic-based method as a tool which can be exploited by the process analysts or system developers to construct models that are accurate and easy to understand through the provision of implicit as well as explicit information on the extensible sets of parameters (concepts) for analysis of the process models at a more conceptual level.

On the other hand, the proposed method references a number of different OWL ontologies, e.g., as shown in Figure 4. Practically in the experimentations described in detail in [9]; for each ontology, all concepts in their turn are all considered by the reasoner (e.g., Pellet) [38] and are checked for consistency by referencing the process parameters within

the underlying ontologies. For instance, based on the behavioural characteristics of the analyzed datasets in [9] which can be found in [14], a cross-validation method was adopted to describe the variability in the composition of the training and test datasets. Moreover, the individual cases (i.e., traces) were computed and recorded according to the reasoner response, and the classification process and evaluations were tested on the resulting outcomes by quantitatively assessing its performance with respect to the correctly classified traces. Thus, for each result of the classification process, the replayable (e.g., true positives – TP) and non-replayable (true negatives – TN) traces were learned. Consequently, the results of the method prove to be more accurate and robust than the conventional process mining and analysis techniques because the method also takes account of the semantic perspectives of the available datasets and models. Moreover, owing to the fact that those models are automatically generated from the actual event logs of the processes in question and carefully integrated with the semantic metrics, the method tends not to unnecessarily lose or leave out important information or missing data.

In general, the main discussion of the results is that as a collection of concepts and predicates, the method has the ability to perform logic reasoning and bridge underlying relations beneath the event logs and the process models with rich semantics. In essence, whenever an inference (semantic reasoning) is made, a generalized association of the process elements is created. Thus, providing consistency checking and analysis of those predicates by tuning the unlabelled apriori models into one (semantic model) that have the best consistency or formal structure. Thus, the term conceptualization.

In theory, the main benefits of the semantic-based annotation and ontological modelling method as described in this paper can be summarized in two forms:

(i) encoding knowledge about specific process domains, and

(ii) contextual analysis and reasoning of the processes at a more abstraction (conceptual) level.

Table 2 is a thematic summary of the main components and supporting tools utilized to demonstrate the real-time implementation of the proposed method of this paper. The different tools were used for annotation and modelling (analysis) of the processes, which can also be applied to any given dataset(s) irrespective of the process domain as long as there is available events log from the process in question, and the extracted event data logs contain the basic minimum requirement for any process mining and analysis task [1]. Besides, the work in this paper shows that the techniques and tools can be utilized by the process analysts or IT experts as a way of performing useful information retrieval and/or query answering in a more efficient, yet effective way compared to other standard logical procedures.

Furthermore, the work has shown that the performance of the semantic-based approach is not only comparable to the outcome of just the process modelling tools and techniques

TABLE 2. Main tools used for implementation of the semantic-based process mining approach

	Main tools
Events Log	Process Logs, e.g., Training Log, Test Log
Process Models	OntoGraph, Fuzzy Models, BPMN Models
Semantic Annotation	Process Description Languages, e.g., OWL, SWRL Rules
OWL Ontology	Protégé Editor, OWLGriD
Reasoning	Pellet, OWL API
Fuzzy-BPMN Mining	ProM, Disco
Semantic Model Analysis	DL Queries, Classification.

but also presents a machine-understandable system that is able to induce new knowledge based on previously unobserved behaviours. Moreover, the technique can be exploited for any form of data analysis (or procedures for prediction and discovery of missing information) especially when analyzing large ontology-based systems.

Therefore, in order to evaluate the performance of the proposed semantic-based approach being able to correctly classify and analyze the individual traces within the resultant models, for instance,

(i) given a trace (t) representing the process behaviour (i.e., true positives or allowed traces) or

(ii) trace (t) representing a behaviour not related to the process (true negatives or disallowed traces) in a given set of data [14],

the work evaluates the results of the experimentations as carried out in [9] using the proposed method of this paper. Characteristics of the datasets which were used to discover the models (from a training event log representing 10 different real-time business process executions, and a set of test event logs provided for evaluation of the employed process mining approach) are as explained in [14]. The test event logs represent part of the original model with a complete total of 20 traces for each of the test logs and are characterized by having 10 traces that can be replayed (allowed) and 10 traces that cannot be replayed (disallowed) by the model. Therefore, a wide variety of problems and analysis are represented. The work has used the test event logs with a complete total of 200 traces to validate the semantic-based method.

Accordingly, the outcomes of the experimentation and cross-validation method were carried out and evaluated alongside other existing benchmark algorithms namely, Inductive Miner and Decomposition [39], DrFurby Classifier [40], Heuristic Alpha+ Miner [41] Fuzzy-BPMN miner [42] that use the same event logs in [14] to discover the process models and replaying semantics of the traces within the event logs.

To do this, the work makes use of the standard Percent of Correct Classification (%PCC) [43] to assess the performance of the different methods. Henceforth, the standard Percent of Correct Classification [43] for the different methods and comparison between the other benchmark algorithms is defined as follows:

$$\text{Log\_PCC} = (\text{number of correctly classified traces})/(\text{total number of traces}) \times 100$$

For example, using the discovered models in the existing Fuzzy-BPMN miner [42], and the proposed Semantic Fuzzy Miner, Table 3 represents an example of how the standard

TABLE 3. Standard Percent of Correct Classification (%PCC) for the test logs and training logs

Model	**Fuzzy-BPMN Miner (%PCC)**	**Semantic-Fuzzy Miner (%PCC)**
Training_Model_1	$(20)/(20) \times 100$ $= 1 \times 100$ $= 100\%$	$(20)/(20) \times 100$ $= 1 \times 100$ $= 100\%$
Training_Model_2	$(16)/(20) \times 100$ $= 0.80 \times 100$ $= 80\%$	$(20)/(20) \times 100$ $= 1 \times 100$ $= 100\%$
Training_Model_3	$(12)/(20) \times 100$ $= 0.60 \times 100$ $= 60\%$	$(20)/(20) \times 100$ $= 1 \times 100$ $= 100\%$

Percent of Correct Classification (%PCC) for the events logs and models were calculated as follows.

Therefore, by using the calculations as shown in Table 3 (standard Percent of Correct Classification (%PCC)) [43] the work outlines in Table 4, the outcome of the semantic-based method against the other existing benchmark algorithms [39-42] in order to weigh up the proposed method and the experimental results. The outcome of the experiments and classification results are as shown in Table 4 and Figure 5.

Consequently, from the evaluation results (see Table 4 and Figure 5), the work notes that the semantic-based annotation and ontological modelling method (Semantic Fuzzy

TABLE 4. The experimental results of the semantic-based method (Semantic-Fuzzy Miner) and other benchmark approaches

	Inductive Miner	**Decom-position**	**DrFurby**	**Fuzzy-BPMN**	**Semantic-Fuzzy**
Model_1	100	100	100	100	100
Model_2	100	100	100	80	100
Model_3	60	95	100	60	100
Model_4	100	100	100	85	100
Model_5	95	100	100	100	100
Model_6	85	95	100	55	100
Model_7	100	100	100	95	100
Model_8	75	70	95	85	100
Model_9	100	100	100	100	100
Model_10	100	100	100	95	100
Ave. Mean – PCC (%)	91.5	96	99.5	85.5	100
No. of correctly classified traces	183	192	199	171	200

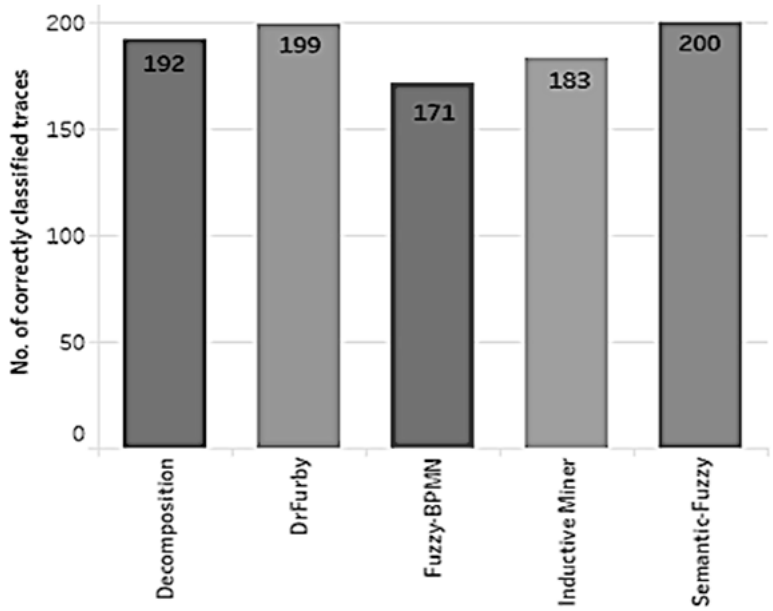

FIGURE 5. Total number of traces correctly classified by each algorithm

Miner) considerably outperforms respectively the Inductive miner [39] and Fuzzy-BPMN miner [42], although the algorithms Decomposition [39] and DrFurby [40] stand for the state-of-the-art classifiers amongst the existing process mining techniques especially when compared to analysis of the results and outcome of the classification process. Additionally, the semantic fuzzy miner has shown an error-free performance when measured using the following classifier formula [1]. Thus,

$$\text{Error} = (\text{fp} + \text{fn})/\text{N}$$

where fp = 0 and fn = 0, such that,

$$\text{Error} = (0 + 0)/200 = 0.$$

Also, the method has shown a high level of accuracy and performance through the following formula:

$$\text{Accuracy} = (\text{tp} + \text{tn})/\text{N}$$

where tp = 100 and tn = 100, such that:

$$\text{Accuracy} = (100 + 100)/200 = 1.$$

In summary, going by the experimental results and validation scores, the precision and recall of the semantic-based method and the classifications process are evidently efficient when compared to the other methods for process mining and analysis.

6. **Conclusion and Future Work.** This paper applies semantic technologies (e.g., semantic annotation, ontologies, and reasoner) to providing formal structures on how to perform and represent the process mining and modelling of any given process domain in a more efficient and accurate manner. The method is proposed to abstract key information that is used to model the relationships that exist between the different process instances that can be found within the knowledge-bases or models. This is done to show how to resolve the different challenges as it concerns semantics aspects that most of the process mining techniques lack. In essence, this paper provides a method that focuses on finding useful structures for the process models and an effective way to analyze and/or determine the relationships that exist within the process knowledge-base.

In theory, the work provides an ontology-based system that is capable of semantically analyzing the different components of the discovered process models. This is owing to the fact that the method is capable of accurately classifying in a formal way (taxonomies) the individual components (classes, objects, and data types) to predict behaviours of unobserved instances (or individual elements) that can be found within the models. This is achieved as a result of making use of the reasoner to carry out the consistency checking, thereby increasing predictive accuracy and automation of the classifications process, and even more, provides an error-free process analysis and performance.

The early studies have shown that semantic technologies (as described in this paper) can be utilized not only to determine the presence of different patterns within the existing models but are also useful towards automatic classification/analysis of the models [44-48]. For example, Wang and Wang [44] provide a method for detecting patterns within the existing knowledge-bases by training a support vector machine (SVM) classifier based on behavioural features of the process instances. Whereas, Janicki et al. [45] propose a framework for the representation of relational structures within the models by identifying traces that are behaviourally equivalent to the observed action sequences. Buhmann et al. [46] propose a novel approach for optimization under uncertainty that proves useful for measuring similarities between the process instances in a given domain, while Tan et al. [47] define a control system that allows for sufficient conditions to be derived in order to guarantee stability and output of the variables. The method described in Zhang et

al. [48] supports a stable approximation approach that is capable of converting the approximation problems to a convex optimization one. Thus, such automated approach to solving the several complex problems and processes is allied to the hybrid intelligent systems (HIS). Interestingly, in the context of the work done in this paper, Jindal and Shweta [49] introduce a modified lexical-semantics based knowledge discovery process that consists of text document collection, data preprocessing, lexical analysis or scanner, semantic analysis, classification, ranking of labels and knowledge discovery. Whereas, Li et al. [50] propose a recurrent neural network model that automatically detects semantic similarity between concepts by supporting a fine-grained reasoning over equivalence or contradiction of pairs of words and phrases. In fact, the aforementioned works/affirmations target one thing in common which is the need for intelligent data/process mining systems or algorithms that are capable of automatically recognizing patterns from the different processes or knowledge-base which they are used to support [51]. Moreover, a typical example of the practical implementation of such a method is the work done by Koga et al. [52] that proposes a new approach for edge-preserving smoothing filter which reconstructs the resulting image locally from the gradient-domain based on a direct method. Besides, this work shows that by understanding and leveraging the real meaning (semantics) of the different process elements which are stored in different variable forms in the datasets or models, the results can be used to identify patterns that can be transliterated into actionable plans and/or process-related decision making in general [53,54].

Moreover, this work proposes the sets of semantically motivated algorithms to realize the following aforementioned contribution which focuses on semantical modelling of the process models, and yet, supports the development of a semantic process mining technique that exhibits a high level of semantic reasoning and capabilities.

Practically, the work makes use of the case study of the learning process to illustrate the capabilities of the proposed technique and its usefulness/application in real time. Indeed, the purpose of designing such an intelligent system is to support (conceptual) analysis of the captured datasets and discovered models capable of providing real-world answers that are closer to human understanding. In other words, the paper focused on the provision of a machine-understandable system rather than just a machine-readable system.

In summary, in addition to the aforementioned contributions and goals of this paper, the work assumes to have presented the main components of such a semantic-based information retrieval, extraction and processing system by not only showing how it integrates the main building blocks (i.e., the use of semantic annotation, ontology, and reasoner) for the semantic-based process mining and analysis but also supports the development and implementation of the methods through the sets of semantically motivated algorithms and series of experimentations.

Nonetheless, one of the limitations of this study is that whilst the paper has presented a set of descriptive algorithms and a conceptual method of analysis to resolve the aforementioned challenges with process mining, there could be potentially many ways to address those problems, or even, bigger areas that have not been yet addressed. This is owing to the fact that the semantic-based process mining is a new area within the process mining field, and there are not too many tools or algorithms that support such an approach currently in the literature. Moreover, there are no current tools capable of directly converting the fuzzy models into some other process modelling formats or notation. As a consequence, the work leverages a varied range of events log conversion such as the BPMN in order to achieve the different viewpoints about the domain processes.

Future works could focus on extending the method through the provision of tools or method capable of automatically integrating the metrics/conversions of the fuzzy models

to other notations in order to support their analysis as well as guarantee the resulting outcomes.

Acknowledgement. The authors would like to acknowledge the technical and financial support of Writing Lab, TecLabs, Tecnologico de Monterrey, in the publication of this work.

REFERENCES

[1] W. M. P. Van der Aalst, *Process Mining: Data Science in Action*, 2nd Edition, Springer-Verlag Heildelberg, Berlin, 2016.

[2] D. Dou, H. Wang and H. Liu, Semantic data mining: A survey of ontology-based approaches, *Proc. of the 9th IEEE International Conference on Semantic Computing*, CA, USA, pp.244-251, 2015.

[3] A. K. A de Medeiros, W. M. P. Van der Aalst and C. Pedrinaci, Semantic process mining tools: Core building blocks, in *Proc. of ECIS*, W. Golden, T. Acton, K. Conboy, H. van der Heijden and V. K. Tuunainen (eds.), Galway, Ireland, 2008.

[4] K. Okoye, A. R. H. Tawil, U. Naeem, S. Islam and E. Lamine, Semantic-based model analysis towards enhancing information values of process mining: Case study of learning process domain, in *Advances in Intelligent Systems and Computing*, A. Abraham et al. (eds.), Springer International, 2018.

[5] W. M. P. Van der Aalst, A. J. M. M. Weijters and L. Maruster, Workflow mining: Discovering process models from event logs, *International Journal of IEEE Trans. Knowledge and Data Engineering*, vol.16, no.9, pp.1128-1142, 2004.

[6] N. Balcan, A. Blum and Y. Mansour, Exploiting ontology structures and unlabeled data for learning, *Proc. of the 30th International Conference on Machine Learning*, Atlanta, GA, USA, pp.1112-1120, 2013.

[7] A. Polyvyanyy, C. Ouyang, A. Barros and W. M. P. Van der Aalst, Process querying: Enabling business intelligence through query-based process analytics, *Decision Support System*, vol.100, no.1, pp.41-56, 2017.

[8] K. Okoye, A. R. H. Tawil, U. Naeem and E. Lamine, Discovery and enhancement of learning model analysis through semantic process mining, *International Journal of Computer Information Systems and Industrial Management Applications*, vol.8, pp.93-114, 2016.

[9] K. Okoye, U. Naeem and S. Islam, Semantic fuzzy mining: Enhancement of process models and event logs analysis from syntactic to conceptual level, *Int. J. of Hybrid Intelligent Systems (IJHIS)*, vol.14, nos.1-2, pp.67-98, 2017.

[10] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider and L. A. Stein, *OWL Web Ontology Language Reference*, Technical Report W3C Proposed Recommendation, Manchester, UK, 2004.

[11] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof and M. Dean, *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*, W3C Member Submission, 2004, http://www.w3.org/Submission/SWRL/, Accessed in February 2019.

[12] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi and P. F. Patel-Schneider, *Description Logic Handbook: Theory, Implementation, and Applications*, 1st Edition, Cambridge University Press, New York, USA, 2003.

[13] A. Rozinat, *Top 5 Data Quality Problems for Process Mining*, Fluxicon, Eindhoven, The Netherlands, 2016, https://fluxicon.com/blog/2011/06/data-quality-process-mining/, Accessed in October 2019.

[14] J. Carmona, M. de Leoni, B. Depair and T. Jouck, *IEEE CIS Task Force on Process Mining Process Discovery Contest @ BPM 2016*, 1st Edition, 2016, http://www.win.tue.nl/ieeetfpm/doku.php?id=shared:edition_2016, Accessed in March 2019.

[15] A. Rozinat, *Process Mining: Conformance and Extension*, Ph.D. Thesis, Technische Universiteit Eindhoven, Eindhoven, the Netherlands, 2010.

[16] K. Okoye, S. Islam, U. Naeem, M. S. Sharif, M. A. Azam and A. Karami, The application of a semantic-based process mining framework on a learning process domain, in *Intelligent Systems and Applications: Proc. of the 2018 Intelligent Systems Conference (IntelliSys)*, K. Arai, S. Kapoor and R. Bhatia (eds.), Cham, Springer, 2019.

[17] F. Lautenbacher, B. Bauer and S. Forg, Process mining for semantic business process modeling, *The 13th Enterprise Distributed Object Computing Conference Workshops*, Auckland, pp.45-53, 2009.

[18] F. Lautenbacher, B. Bauer and C. Seitz, Semantic business process modeling – Benefits and capability, *AAAI Spring Symposium: AI Meets Business Rules and Process Management*, Stanford University, CA, USA, 2008.
[19] M. Born, F. Dörr and I. Weber, User-friendly semantic annotation in business process modeling, in *Web Information Systems Engineering – WISE 2007 Workshops*, M. Weske, M. Hacid and C. Godart (eds.), Berlin, Heidelberg, Springer, 2007.
[20] C. Jonquet, A. Toulet, B. Dutta and V. Emonet, Harnessing the power of unified metadata in an ontology repository: The case of AgroPortal, *Journal on Data Semantics*, pp.1-31, 2018.
[21] A. K. A. de Medeiros and W. M. P. Van der Aalst, Process mining towards semantics, in *Advances in Web Semantics*, T. Dillon, E. Chang, R. Meersman and K. Sycara (eds.), Berlin, Heidelberg, Springer, 2009.
[22] W. Jareevongpiboon and P. Janecek, Ontological approach to enhance results of business process mining and analysis, *Journal of Business Process Management*, vol.19, no.3, pp.459-476, 2013.
[23] K. Okoye, A. R. H. Tawil, U. Naeem and E. Lamine, Semantic reasoning method towards ontological model for automated learning analysis, in *Advances in Intelligent Systems and Computing*, N. Pillay, A. Engelbrecht, A. Abraham, M. du Plessis, V. Snášel and A. Muda (eds.), Switzerland, Springer, 2016.
[24] D. Calvanese, T. E. Kalayci, M. Montali and S. Tinella, Ontology-based data access for extracting event logs from legacy data: The onprom tool and methodology, in *Business Information Systems*, W. Abramowicz (ed.), Cham, Springer, 2017.
[25] T. R. Gruber, Toward principles for the design of ontologies used for knowledge sharing, *Int. J. of Human-Computer Studies*, vol.43, nos.5-6, pp.907-928, 1995.
[26] O. O. Petrenko and A. I. Petrenko, A model-driven ontology approach for developing service system applications, *Journal of Computer Science Application Information Technology*, vol.2, no.4, pp.1-7, 2017.
[27] K. Okoye, A. R. H. Tawil, U. Naeem, S. Islam and E. Lamine, Using semantic-based approach to manage perspectives of process mining: Application on improving learning process domain data, *Proc. of the 2016 IEEE International Conference on Big Data (BigData)*, Washington, D.C., pp.3529-3538, 2016.
[28] C. d'Amato, N. Fanizzi and F. Esposito, Query answering and ontology population: An inductive approach, in *Proc. of the 5th Euro. Semantic Web Conference*, S. Bechhofer, M. Hauswirth, J. Hoffmann and M. Koubarakis (eds.), Berlin, Heidelberg, Springer, 2008.
[29] G. Antoniou, P. Groth, F. van Harmelen and R. Hoekstra, *A Semantic Web Primer*, 3rd Edition, The MIT Press, Cambridge, Massachusetts London, England, 2012, www.aryabarzan.info/slides/a_semantic_web_primer.pdf, Accessed in February 2019.
[30] J. Lehmann and P. Hitzler, Concept learning in description logics using refinement operators, *Machine Learning*, vol.78, nos.1-2, pp.203-250, 2010.
[31] M. M. Rahman and T. W. Finin, Understanding the logical and semantic structure of large documents, *CoRR*, abs/1709.00770, 2017.
[32] M. Sabou, L. Aroyo, K. Bontcheva, A. Bozzon and R. K. Qarout, Semantic web and human computation: The status of an emerging field, *Semantic Web (SWJ)*, vol.9, pp.291-302, 2018.
[33] L. Zadeh, Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets and Systems*, pp.3-28, 1978.
[34] L. Zhao and R. Ichise, Ontology integration for linked data, *Journal on Data Semantics*, vol.3, no.4, pp.237-254, 2014.
[35] M. Pfaff, S. Neubig and H. Krcmar, Ontology for semantic data integration in the domain of IT benchmarking, *Journal on Data Semantics*, vol.7, no.1, pp.29-46, 2018.
[36] G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi and R. Rosati, Using ontologies for semantic data integration, in *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years. Studies in Big Data*, S. Flesca, S. Greco, E. Masciari and D. Saccà (eds.), Cham, Springer, 2018.
[37] D. C. Wimalasuriya and D. Dou, Ontology-based information extraction: An introduction and a survey of current approaches, *Journal of Information Science*, vol.36, no.3, pp.306-323, 2010.
[38] E. Sirin and B. Parsia, Pellet: An OWL DL reasoner, *Proc. of the 2004 International Workshop on Description Logics (DL2004)*, Whistler, British Columbia, Canada, 2004.
[39] R. Ghawi, *Process Discovery Using Inductive Miner and Decomposition*, In CoRR abs/1610.07989 (2016) Technical Report Submission for the Process Discovery Contest @ BPM 2016, 1st Edition, https://arxiv.org/abs/1610.07989, Accessed in May 2019.

[40] E. Verbeek and F. Mannhardt, *DrFurby Classifier: Process Discovery Contest @ BPM 2016*, Technical Report Submission for the Process Discovery Contest, 1st Edition, 2016, http://www.win.tue.nl/∼hverbeek/wp-content/uploads/2016/05/drfurby12.pdf, Accessed in May 2019.

[41] M. Shteiner, L. Bodaker and A. Senderovich, *Heuristic Alpha+ Miner (HAM): Process Discovery Contest 2016*, Technical Report Submission for the Process Discovery Contest @ BPM 2016, 1st Edition, 2016, https://web.iem.technion.ac.il/images/ISE-TR-16-2.pdf, Accessed in May 2019.

[42] K. Okoye, U. Naeem, S. Islam, A. R. H. Tawil and E. Lamine, Process models discovery and traces classification: A fuzzy-BPMN mining approach, *J. of International Technology and Information Management*, vol.26, no.4, pp.1-50, 2017.

[43] K. Baati, T. M. Hamdani, A. M. Alimi and A. Abraham, Decision quality enhancement in minimum-based possibilistic classification for numerical data, in *Advances in Intelligent Systems and Computing (AISC)*, A. Abraham, A. K. Cherukuri, A. M. Madureira and A. K. Muda (eds.), Springer International, 2018.

[44] P. Wang and Y. Wang, Malware behavioural detection and vaccine development by using a support vector model classifier, *Journal of Computer and System Sciences*, vol.81, no.6, pp.1012-1026, 2015.

[45] R. Janicki, J. Kleijn, M. Koutny and L. Mikulski, Classifying invariant structures of step traces, *Journal of Computer and System Sciences*, vol.104, pp.297-322, 2019.

[46] J. M. Buhmann, A. Y. Gronskiy, M. Mihalák, T. Pröger, R. Šrámek and P. Widmayer, Robust optimization in the presence of uncertainty: A generic approach, *Journal of Computer and System Sciences*, vol.94, pp.135-166, 2018.

[47] H. Tan, Z. Huang and M. Wu, Data-based predictive control for networked non-linear multi-agent systems consensus tracking via cloud computing, *IET Control Theory & Applications*, vol.13, no.5, pp.683-692, 2019.

[48] X. Zhang, M. Li, H. Ding and X. Yao, Data-driven tuning of feedforward controller structured with infinite impulse response filter via iterative learning control, *IET Control Theory & Applications*, vol.13, no.8, pp.1062-1070, 2019.

[49] R. Jindal and Shweta, A modified knowledge discovery process in the text documents, *International Journal of Innovative Computing, Information and Control*, vol.14, no.3, pp.817-832, 2018.

[50] X. Li, C. Yao, Q. Zhang and G. Zhang, Semantic similarity modeling based on multi-granularity interaction matching, *International Journal of Innovative Computing, Information and Control*, vol.15, no.5, pp.1685-1700, 2019.

[51] B. Huynh, C. Trinh, V. Dang and B. Vo, A parallel method for mining frequent patterns with multiple minimum support thresholds, *International Journal of Innovative Computing, Information and Control*, vol.15, no.2, pp.479-488, 2019.

[52] T. Koga, S. Furukawa, N. Suetake and E. Uchino, Edge-preserving smoothing based on local gradient-domain processing, *ICIC Express Letters*, vol.11, no.7, pp.1175-1183, 2017.

[53] Z. Yu, A. Abraham, X. Yu, Y. Liu, J. Zhou and K. Ma, Improving the effectiveness of keyword search in databases using query logs, *Engineering Applications of Artificial Intelligence*, vol.81, pp.169-179, 2019.

[54] A. Abraham, E. Au, A. Binotto, L. Garcia-Hernandez, V. Marik, F. G. Marmol, V. Snasel, T. I. Strasser and W. Wahlster, Industry 4.0: Quo Vadis?, *Engineering Applications of Artificial Intelligence*, vol.87, 2020.