An Inferential Knowledge Model for the Digitalisation and Automation of Business Process Analysis

Anne Füßl[1], Volker Nissen[1], Franz Felix Füßl[2], and Simon Dopf[1]

[1]TU Ilmenau, FG WI für Dienstleistungen, Ilmenau, Germany; [2]iTech Solutions, Geschwenda, Germany {anne.fuessl,volker.nissen}@tu-ilmenau.de

Abstract. In the course of digitalisation, the service sector is undergoing rapid change, which is partly questioning the existing business models and influencing the conditions of competition. The consulting industry, too, is confronted with corresponding challenges despite the positive sales trend. The use of advanced technology-based tools based on artificial intelligence or analytical approaches could sustainably improve the competitive situation of consulting providers. Consulting projects often require a comprehensive analysis of business processes. The knowledge-based system iKnow presented here has mechanisms of automated inference as well as machine learning approaches. In contrast to process mining, it can also be used to support and partially automate process analysis if no evaluable log data is available. In this article, the previous concept of an inference-capable business process analysis tool is further developed, its usefulness is demonstrated and evaluated using an example process. With the help of a modeled knowledge base for process analysis, BPMN models can be automatically examined for weak points on the basis of analysis criteria and suitable improvement measures can be determined. Machine learning approaches can be used to continuously improve the analysis results.

Keywords: Automated Business Process Analysis, Knowledge Modeling, Machine Learning, Digital Consulting Technologies, Virtualisation of Consulting

15th International Conference on Wirtschaftsinformatik,
March 08-11, 2020, Potsdam, Germany

https://doi.org/10.30844/wi_2020_w1-fuessl

1 Research Motivation in the Context of Digital Consulting Technologies

Digitalisation can be described as a change from real processes and services to information system based services [1]. Today it forms the basis of many innovative business models which aim at the flexibilisation, individualisation and autonomisation of services [2–4]. Bughin et al. [5] defines digitalisation as the ability to network people, devices and physical objects independently of time and place. The improved processing and analysis capability of information from large data sets leads to a higher level of automation of processes and decisions.

Consultancies are faced with the challenge of reacting to digital changes and using new technologies to innovatively transform classic consulting processes and business models and to adapt to the changed consulting market [6]. One possible approach is the virtualisation of consulting processes. This means the personal on-site contact between consultant and client is reduced to a reasonable minimum through the use of technology [7]. The extreme case of a purely technology-based consulting approach can be described as full automation [4, 7]. This requires suitable IT artefacts. The generic tool iKnow presented in this article is based on a formalised knowledge model, mechanisms of automatic inference and machine learning approaches [8]. Due to the knowledge-intensive character of the consulting industry, iKnow can be used specifically for knowledge processing and decision support within consulting projects. In a first project, the analysis of business processes will be considered [9].

In contrast to process mining, it is *not* assumed that the business processes under consideration are IT-supported and that the log data can be evaluated for this purpose. The results of process mining include the reconstruction and analysis of processes in order to determine implicit and hidden process information from data of IT systems [10]. In contrast to business process simulation, no simulation-relevant process data is required. The simulation usually supports the assessment of dynamic process behavior and achieves quantitative time- and cost-related result data, which concern the improvement of the interaction between human-task-technology [11]. Neither simulation nor process mining consider soft and qualitative influencing factors such as implicit process knowledge of the process participants. Without supplementary procedures, they also do not allow automatic interpretation of the results obtained or decisions on process improvements [9]. In contrast, iKnow should serve as the basis for a knowledge-based and inferentially capable business process analysis tool and enrich the automated process analysis with context-related information and expert knowledge on the respective process.

In this article, the previous concept is further developed and the evaluation of iKnow for automated business process analysis in the context of consulting tasks is presented. On the basis of defined analysis criteria, Business Process Model and Notation (BPMN) models can be examined for weak points and then suitable improvement measures can be determined automatically. With the help of iKnow's

https://doi.org/10.30844/wi_2020_w1-fuessl

automated learning approaches, context-related process information and expert knowledge can be integrated so that the automated business process analysis can be continuously improved over time.

Methodically, the processing of the contribution can be assigned to design oriented business informatics. The design science research approach according to Peffers et al. [12] is chosen as the framework. After the description of the research motivation, the basics of iKnow as well as core elements of the existing concept of the business process analysis tool are presented. The formal model in iKnow is adapted and extended in chapter 3 according to the requirements of an automated business process analysis. The methodological development of the artifact is based on a formal-deductive analysis [13]. Considering the limited space, the main focus of this paper will be on the conceptual design and in particular on the demonstration of the feasibility and usefulness of the artifact for automated business process analysis. Finally, a prototype evaluates the designed tool using an example of a business process. The prototype itself and the evolutionary procedure for developing the prototype application are not described. The last chapter sums up the findings gained and points out subsequent research work.

2 iKnow – a Tool for Automated Business Process Analysis

2.1 Foundations of the Knwoledge Model Behind iKnow

iKnow contains a generic abstract model for the formal representation of knowledge with the aim of creating information and solving problems [8]. Deduction algorithms with the consideration of variable influencing factors enable the formation of inferences and thus logical reasoning on the basis of modelled knowledge. Using machine learning approaches, new knowledge can be acquired and formally represented in this way. The IT-based implementation of the model iKnow can be assigned to the system class of knowledge-based expert systems [14]. It consists of a knowledge base, inference mechanisms in the form of deduction algorithms and an explanation facility, which is closely linked to the four-level architecture of the model. In the following, some of the most important basics of the model are explained. A comprehensive description of the development of iKnow can be found in [8].

Elements and Associations. The iKnow architecture consists of four levels: input layer (L4), data transfer layer (L3), information processing layer (L2) and abstraction layer (L1). In the form of a directed and edge-weighted graph, the elements of the layers are connected by different associations and rules. On the basis of association rules, automated inference formation is made possible and suitable methods of machine learning can be integrated [4].

In the application layers (L4 and L3), the elements *Feature* and *Data Source* serve to communicate and integrate data and information, such as business process models. Data Source elements (L4) collect data via a simple request-response principle from external sources, such as measuring instruments or sensors, or from questions to the

https://doi.org/10.30844/wi_2020_w1-fuessl

system user. Feature elements (L3) represent influencing factors on a modelled knowledge base (e.g. share of manual activities, compatibility between IT systems or staff motivation). *Cells* (L2) and *Items* (L1) collect data by analyzing their associations with neighboring elements using deduction algorithms and transforming them into information. Thus they represent the basis for the generation of knowledge. Items represent a descriptive view of the modelled knowledge, Cells on the other hand represent concrete instances. The Item *IT System* can, for example, be connected to the Cell *SAP ERP* as a concrete form via an is association (*SAP ERP* is IT System). Tasks, functions or activities can be represented via the *Activity* element type. A combination of a Cell (e.g. *process*) and an Activity (e.g. *document*) can be modeled by a *Combining* (e.g. *document process*), as well as two Cells or Items (e.g. *process analysis* or *optimised process*). Thus, combinings can be used, for example, to represent process activities within process models. Cells or Combinings, which consist of two Cells, can be used to describe any process event, resource or operator (e.g. *request confirmed*, *ordering system* or *service provider*). The modelling regulation of the individual elements can be found in Chapter 2.2, Fig. 1. [8]

The associations between the individual elements or nodes in the knowledge model are formalized using defined association classes. These association classes can be used uniformly by the deduction algorithms and enable the formation of inferences within a knowledge model. The following five basic association classes serve for knowledge representation and were developed by Füßl [8]: *is*, *has*, *can*, *part-of* and *used-for*. The use of individual associations is illustrated during the development of the business process analysis tool in Chapter 3.

Deduction Algorithms and Machine Learning. Each element has a result. This is influenced by all adjacent element nodes and their result generation. An element result is generated by calculating a Boolean expression. In the model, this is represented by the design of *Constraints*. The goal of the constraints is to make the inference in the knowledge model more efficient by processing elements step-by-step through as few path runs as possible [8].

A constraint can take the Boolean values *true* (1) or *false* (0). The corresponding calculation formula is called *Result Expression* and is defined in the elements of levels L1 and L2 (Items, Cells, Activities or Combinings). If an element assumes the value *false*, this node is excluded from the deduction algorithms for the further process steps of inference formation. This type of node reduction during result generation is called Deductive Reasoning Element Pruning (DREP). This efficiently reduces the number of elements to be passed through during result generation [8].

The basic architecture of iKnow offers two views of a knowledge model: a descriptive view, which contains explicit knowledge, and a deciding view, which generates knowledge on the basis of variable influencing factors. Taking influencing factors into account, the relevant explicit Knowledge Elements can be identified and new Knowledge Elements derived. Feature elements represent such influencing factors and are determined by Data Source elements. Deduction is based on different algorithms. The elements and association classes of the model form the basis of the deduction algorithms. Three of the five main algorithms according to Füßl [8] (*Isn't-*

https://doi.org/10.30844/wi_2020_w1-fuessl

it?, *Kind-of?* and *Find!*) are presented in chapter 3 as part of the development of the business process analysis tool.

Moreover, the integration of questions to the system user is highlighted as a machine learning component here. When executing the *Isn't-it?* or *Kind-of?* algorithm, conflicts can arise in the result. The deduction of a Feature can generate different elements as a solution set due to several constraints. Either no unique solution element has been identified or conflicting elements have arisen in the solution set. *Interaction with an expert user* can help solve this problem. Thus, by creating a new Knowledge Element, a user interaction can request confirmation of all linked Knowledge Elements identified based on Feature transfers and deduction algorithms. The user interaction with an expert in this context provides the possibility to model further Knowledge Elements [8].

Furthermore, in the development of the business process analysis tool, the technical *transfer of Features from abstractions to concretisations* was taken into account as a machine learning approach [15], which will however not be described in more detail in this article for reasons of space.

2.2 Overview of Adaptation for Automated Business Process Analysis

The starting point for the business process analysis is the design of a knowledge model according to the formal regulation of the abstract model behind iKnow ($G = (V, E, i_G)$, Figure 1).

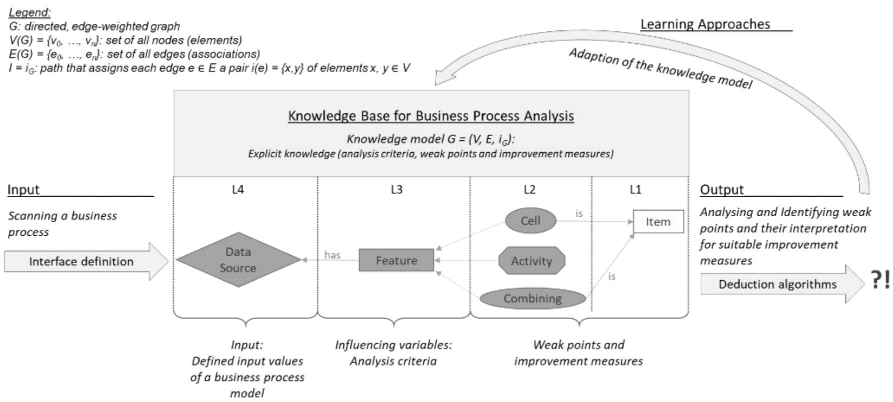

Figure 1. Rough concept of the business process analysis tool (following [9])

Based on business process objectives, it is necessary to model deficiency indicators as analysis criteria, resulting weak points of processes and suitable improvement measures as explicit information in the form of a knowledge base. The focus for the analysis of business processes is currently on BPMN 2.0 models, since the notation rule has differentiated modeling elements for mapping processes, and is widespread as well as universally applicable due to its XML structure [16]. The input data required for business process analysis can be automatically determined from a BPMN file via a defined interface, which is to be implemented in level L4 as a Data Source element (Figure 1, left). The analysis criteria are modeled in the form of Feature elements and thus influence the results generated by deduction algorithms. The generated output includes possible weak points or suitable improvement measures for the analysed process. The machine learning approaches provided in the abstract model of iKnow can over time lead to an adaptation and modification of the existing knowledge model with each analysed process model. In this way, the knowledge base for the analysis of business processes can be constantly improved, which implies an improvement in the automated generation of results. This means that iKnow is capable of learning. Fig. 1 illustrates the rough concept. More details on the general model mechanisms and adaptations can be found in [8] and [15]. The next chapter highlights the advancement of iKnow from an essentially generic reasoning tool to an instrument for automated business process analysis.

3 Design of the Business Process Analysis Tool

3.1 Development of the Knowledge Base for the Analysis of Business Processes

For automated business process analysis with iKnow, all the information on which the business process elements are based is used in conjunction with the existing knowledge of a given iKnow knowledge base. Generic knowledge structures are thus integrated into the inference formation and the results of a process analysis can be taken into account in subsequent analyses. The basis of the business process analysis tool to be developed is a knowledge model $G_{process\_analysis}$ according to the formal description of iKnow. For this, it is necessary to design explicit knowledge for business process analysis as a generic iKnow knowledge model. The scope and correctness of modeled possible process weaknesses and potentially suitable improvement measures has a direct influence on the quality of the analysis results. Therefore, the integration of new Knowledge Elements and the expansion of the knowledge base should be possible at any time. In the following, the development of the knowledge base is explained exemplarily for the process weak point *media break* with the corresponding analysis criteria. An overview of potential other weak points and associated analysis criteria for processes can be found in [9]. For reasons of space, the description of knowledge modelling for improvement measures must be omitted in this article. Within the framework of the analysis in section 3.2, however, this can be seen in the results.

To model weak points in an iKnow knowledge model, elements of levels L1 and L2 are used. For classification, these require a superordinate Item called *Weak Point*, which is the direct or indirect *is*-successor of individual process flaws, such as *data redundancy*, *organizational break*, *system break* and *media break* (Figure 2). The element *optimised process* is represented by a Combining and consists of the Item

https://doi.org/10.30844/wi_2020_w1-fuessl

process and the descriptive Knowledge Element *optimised* as a Cell. The constraint $\gamma_1(e_1(v_{optimised\_process}, v_{weak\_point})):r(v_{weak\_point})==$ false) indicates whether a process has improvement potential and is represented in Figure 2 by a dashed line between the nodes $v_{optimised\_process}$ and v_{weak_point}. If there is a weak point in the business process to be analysed, not only the corresponding Combining (here: *data redundancy*, *media break*, *system break* or *organizational break*) has the value true, but also all *is*-successors. Accordingly, the Item *weak point* is true if a flaw has been identified in the analysed business process. If, for example, the Combining *media break* becomes *true*, the result value *true* also results for the *is*-successor elements *break* and *weak point*. Using the constraint γ_1 described above, the result of the node $v_{optimised\_process}$ becomes true if the result of the node $r(v_{weak\_point})$ is false. Thus, the Combining *optimised process* is connected to all flaws present in the process model via the constraint γ_1 (node v_{weak_point}, its *is*-predecessor $v_{data\_redundancy}$ and v_{break}, as well as v_{media_break}, v_{system_break} and $v_{organizational\_break}$). In this way, the Combining *optimised process* can be used as an entry point to check whether a business process to be analysed is already an optimised process. For further description, the weak point *media break* is now focused.

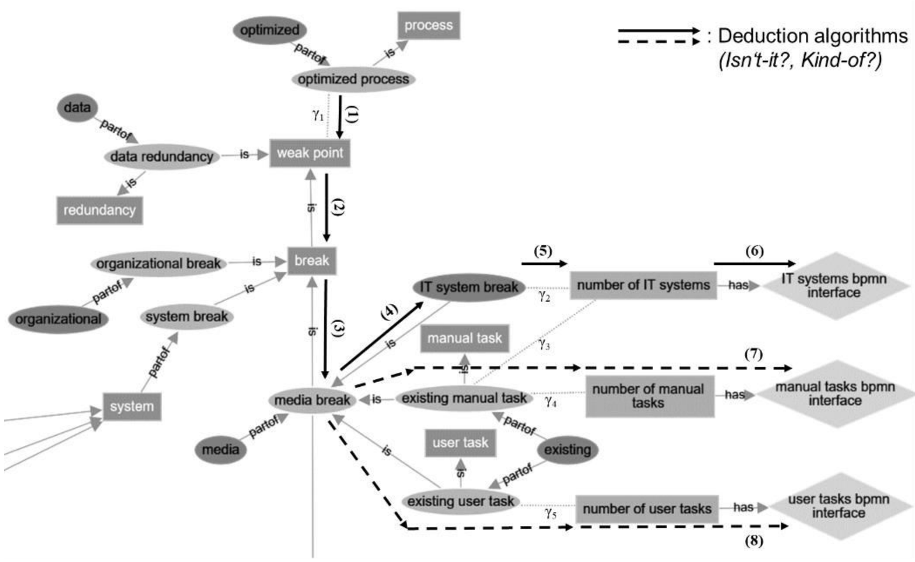

Figure 2. Extract of the knowledge model $G_{process\_analysis}$ with labelling of the deduction algorithms *Isnt-it?* and *Kind-of?*

To model analysis criteria and define them, the iKnow elements of level L3 are required. Features serve as influencing variables in the knowledge model and thus represent conditions when a weak point in the process is detected. Feature elements must be linked with defined Data Source elements (L4), since they require the respective input parameters (input values) of the process to be analysed. Using a

https://doi.org/10.30844/wi_2020_w1-fuessl

defined calculation formula (*Result Expression*), a result value can be determined for each analysis criterion on the basis of the required input values. To enter automated business process analysis, the BPMN models to be analysed must be properly modelled according to their notation rules. Each Data Source element addresses a BPMN model to be analysed via a defined interface and expects the input values required for the respective analysis criterion. Since each BPMN model is available as a standardised XML file, the query language XML Path Language is used to design the functionality of the interfaces [17, 18]. The definitions of the interfaces are not described in this article for reasons of space. Figure 2 shows three Data Source elements, represented as diamonds. The Feature elements are linked to the Data Sources via *has*-associations. The dashed edges between the Feature elements and the elements *IT system break*, *existing manual task* and *existing user task* indicate existing constraints. As soon as a condition of these elements is fulfilled, the *is*-successor *media break* also receives the value *true*. The media breaks shown in Figure 2 are identified by the Features and their Result Expression as follows:

- If two different IT systems are in a business process as non-integrated systems, a media discontinuity is discernible: $r(v_{number\_of\_IT\_systems}) > 1$. The prerequisite for this is correct modeling in the BPMN model. Non-integrated systems must always be modelled as separate pools according to the notation specification of BPMN. In case of doubt, the necessary information must be obtained from the user via a user question (Chapter 3.3, Inductive Learning Approaches).
- According to BPMN 2.0, a user task exists as soon as a human is in dialog with an IT system [16]. Every user task leads to a media break if data input and data processing are based on two different media types. For instance documents, which will be manually transmitted to a processing IT system. So, if at least one user task is modeled in a business process, there is a media break: $r(v_{number\_of\_user\_tasks}) > 0$. For a better understanding, user tasks with data input directly into integrated IT systems, e.g. online forms of ERP-systems, will not be further considered in this paper.
- Nevertheless, a media discontinuity in processes that also have manual tasks in addition to an IT system must be identified. Accordingly, this form of media disruption is influenced by two Feature elements: $r(v_{number\_of\_manual\_tasks}) > 0$ AND $r(v_{number\_of\_IT\_systems}) > 0$.

The analysis results are generated by deductive inference on the basis of deduction algorithms. In the following the business process analysis is demonstrated with reference to four analysis questions.

3.2 Demonstration of Result Generation

I. Is the process at hand already optimised? The first step in the process analysis is the question whether the process model to be examined is already an optimised process. For this simple decision-making process, the *Isn't-it?* algorithm is applied to the Combining *optimised process*. On the basis of the input values of the associated

Feature elements, the solution of Result Expressions as well as the Association Classes and their Element types of a knowledge base, the Combining *optimised process* is examined to determine which result value (*true* or *false*) it has [8]. The analysis of the question comprises six iteration steps (Figure 2). In step (1), the Knowledge Element *optimised process* expects the result value of the Item *weak point*. The result value of the *optimised process* Element is dependent on the *weak point* Element by the constraint γ_1. If no weak point return value exists, the *Isn't-it?* algorithm iteratively passes through all *is*-predecessors and checks the associated Elements for existing result values (steps (2-3)). After Step (3) follow three *is*-predecessors to the Element *media break*. Therefore, step (4) first analyses the result value of the neighboring *is*-predecessor *IT system break*. This result value is influenced by a Result Expression for the Feature Element *number of IT systems*, which is modelled by the constraint $\gamma_2(e_2(v_{IT\_system\_break}, v_{number\_of\_IT\_systems}): r(v_{number\_of\_IT\_systems}) > 1)$ in Figure 2. If no return value from the Result Expression exists in step (4), the input value required for the Result Expression is queried via the Feature Element *number of IT systems* in step (5). Step (6) finally addresses the linked Data Source element of type Interface. Thus, the required input value of the Feature Element is extracted from a linked BPMN process model. All iteration steps now receive their return values in reverse order (steps (6-1)). Depending on whether the Element *weak point* now has the result *true* or *false*, the final result for the entry element *optimised process* correspondingly becomes *false* (no optimised process) or *true* (optimised process):

- no optimised process: $r(v_{optimised\_process}) = \text{true} \leftarrow r(v_{weak\_point}) == \text{false}$ rsp.
- optimised process: $r(v_{optimised\_process}) = \text{false} \leftarrow r(v_{weak\_point}) == \text{true}$.

Starting from step (4), the *Isn't-it?* algorithm sequentially passes each *is*-predecessor until one of them returns the result value *true*. Consequently, steps (7) and (8) are only triggered if one of the neighboring *is*-predecessors of the *media break* Element is not already *true*. This procedure is based on the efficient inference formation in the abstract model iKnow using DREP [8].

II. Which weak points are present in the process? The second analysis question is answered with the help of the *Kind-of?* algorithm is executed on the Knowledge Element *weak point*. The *Kind-of?* algorithm is used to identify all neighboring *is*-predecessors of a node with the result value *true* [8]. First, the Element *weak point* is checked as the entry element, whether the *is*-predecessor break has the result value *true* (step (2)). In step (3), all linked *is*-predecessors of the source element *break* are then analysed. In comparison to *Isn't-it?* algorithm, the *Kind-of?* algorithm, <u>all</u> *is*-predecessors (*IT system break*, *existing manual task* and *existing user task*) are analysed to see whether they have the return value *true*. Therefore, steps (7) and (8) using the *Kind-of?* algorithm are <u>always</u> executed. If a neighboring *is*-predecessor has already been analysed, the existing result value is returned. If no result value exists, steps (5) and (6) are performed to determine the return value. Depending on the respective entry element (in this case: kind-of "*weak point*"), the solution set of the *Kind-of?* algorithm includes all adjacent *is*-predecessor elements with the return value *true*. Thus, if there is a *media break*, the result *break* is returned first. In the case of a follow-up question, which concrete media breaks exist (kind-of "*media break*"), the solution set comprises the Knowledge Elements *IT system break*, *existing manual task* and *existing user task*, provided their result value is *true*.

III. Which improvement measures are fitting? In order to simultaneously receive proposals for solutions to an identified weak point, suitable improvement measures should be identified during process analysis. For this purpose *iKnow* provides the *Find!* algorithm. Regarding an abstract Knowledge Element, concrete instances can be found in the knowledge model. This requires a classification of the sought-after element(s) for process improvement (*improvement measure*), which in turn is linked to concrete suggestions for improvement (e.g. *improve communication*) via *is*-associations. Furthermore, a specification of usability is necessary in order to determine the fulfilment of the purpose of the searched Knowledge Elements [8]. Thus, the improvement measures sought each have an *is*-association with the Combining *improvement measure* and a *used-for*-association with the Combining *handle media break* (*improvement measure used-for handle media break*).

As input information to answer the analysis question three the *Find!* algorithm requires the following Knowledge Elements: Find *improvement measure* to *handle media break*. The two Knowledge Elements are the respective entry points. Starting from the intended purpose *handle media break*, all neighbouring *used-for*-predecessors with the classification *improvement measure* are included in the solution set of the analysis (e.g. *improve communication*, *improve coordination*, *define rules*).

IV. Which aids and tools are suitable? In addition to the suggestions for improvement, suitable aids for the implementation of these measures should also be identified for each process analysis. Again, the *Find!* algorithm is used. With the input information: Find *tool* to *improve communication*, the respective *used-for*-predecessors of the classification *tool* are returned (e.g. *groupware system* and/or *workflow management system*).

According to the described procedure it is possible to add further weak points and associated analysis criteria to the knowledge base and to model suitable improvement measures for each specific weak point as well as suitable aids in the form of expert knowledge in the knowledge model $G_{process\_analysis}$.

iKnow offers further basic algorithms, such as the *Characteristics?* algorithm. Via *has*-associations concrete Knowledge Elements thereby can receive properties (e.g. *groupware system* has *installation type* or *groupware system* has *features*). Thus, the solution sets of the analysis questions three and four can contain additional information to the respective improvement measures and aids [8].

https://doi.org/10.30844/wi_2020_w1-fuessl

3.3 Inductive Learning Approaches

Every analysis of a business process can lead to an adaptation or extension of the knowledge base or the knowledge model $G_{process\_analysis}$. Feature Elements represent the analysis criteria that are necessary to determine weak points. They require certain input values from a business process in the form of an XML file (Chapter 3.1). If process elements contained in the process model are not known to the knowledge model $G_{process\_analysis}$, a question is put to the user of *iKnow*. This form of machine learning can also be called active learning [19, 20]. For the use case of analyzing all IT systems contained in the modelled process, the function defined in the interface is called via the Data Source element *IT systems bpmn interface* (Figure 2). With the help of *Isn't-it?* algorithm the system checks whether the BPMN pools in the process are IT systems known to the $G_{process\_analysis}$ knowledge model. All Knowledge Elements, which are not known in the knowledge model so far, are assigned to the respective element in the knowledge model (e.g. the Combining *IT system*) by a multiple-choice user question. Thus, the *iKnow* user answers the system whether the unknown process elements are e.g. IT systems (see Evaluation, Figure 4). The user therefore acts as an agent of a knowledge-based and learning system [19]. Answering such questions results in further *is*-types of a higher-level element. On the other hand, when Combinings such as *groupware system* are added, it is checked whether a component of the new Combining already exists in the knowledge model. In this way, the duplicate creation of identical Knowledge Elements is prevented and new Knowledge Elements are sensibly linked with existing ones. Consequently, the number of unknown process elements or Knowledge Elements is continuously reduced and the time required for process analysis can be reduced when importing further process models.

In order to ensure and constantly improve the quality of the process analysis, the determined analysis results should, at this stage of development, be examined for their correctness and completeness by a process analysis expert. On the one hand, the expert has the possibility to delete the suggested improvement measures and aids (analysis questions III. and IV., chapter 3.2) from the determined solution set, if these are not suitable for the removal of the determined weak point(s). On the other hand, it should be possible to extend the analysis results and add missing improvement measures or aids by appending them to the knowledge model. In this way, the knowledge model can be continuously improved for future business process analyses [9].

A situation, in which an existing weakness has not been identified or an incorrect weakness has been identified, is more complex. In this case, the source of the error lies in the definition of the Data Source interface, which requires a modification. The exact design of this machine learning component requires a detailed conception and is therefore not the focus of this article.

https://doi.org/10.30844/wi_2020_w1-fuessl

3.4 Evaluation by Prototype

To demonstrate and evaluate the business process analysis tool, a Windows application was constructed as a prototype. Using an editor window as the user interface, the knowledge base required for process analysis can be created as a knowledge model with the help of corresponding input commands. This contains the process weaknesses and analysis criteria developed in Chapter 3.1 as well as exemplary improvement measures and aids. At the beginning of a business process analysis, the created knowledge model $G_{process\_analysis}$ is loaded into the prototype. Subsequently, the analysis of a business process can take place on the basis of the presented analysis questions for the identification of media breaks. As an example process, a simplified order fulfilment process is chosen, see Figure 3.

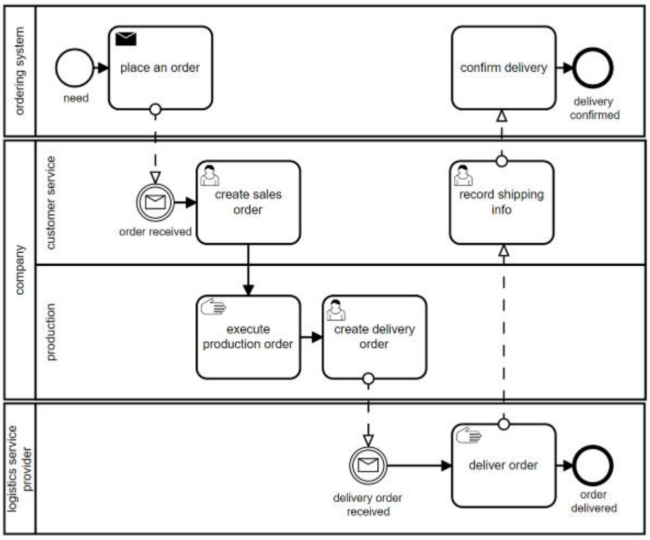

Figure 3. Simplified order fulfillment process as BPMN 2.0 model

The process analysis begins with the first question (I. Is the process already optimsed?) using the following input command:

1. < isnt-it "optimized process" → Answer: NO

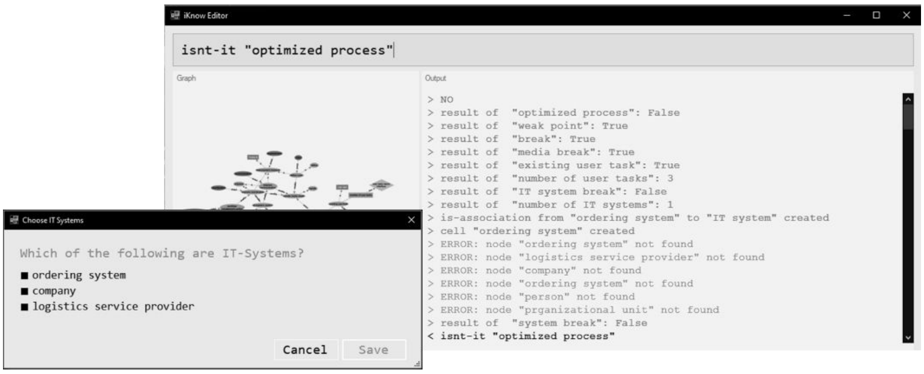

Figure 4. Screenshot for analysed question 1 – Is the process already optimised?

An Explorer window opens to select and load the sample process in BPMN format. According to the iteration steps described in chapter 3.2, it is analysed whether the Knowledge Element *optimised process* has the result value *true* or *false*. Since the three pool names of the loaded knowledge base are not yet known in the example process, the prototype generates a user question to clarify the unknown Knowledge Elements (Figure 4, left). Once the ordering system has been manually selected as the IT system, it is added to the knowledge base for future analysis. Figure 4 (right) shows the logs *cell "ordering system" created* and *is-association from "ordering system" to "IT system" created*. Now the calculations of the Result Expressions take place. The log output shows that this is not an *IT system break* ($r(v_{number\_of\_IT\_systems}) = 1 \rightarrow$ false). The number of existing user tasks is three ($r(v_{number\_of\_user\_tasks}) = 3 \rightarrow$ true). Therefore, the respective *is*-successor elements *media break*, *break* and *weak point* also receive the result value *true*, so that the answer to the initial question is *false* (Figure 4, right, first line: NO).

The second question, which particular flaws are involved in the process, can be examined with the help of the following input commands:

2. < *kind-of* "weak point" → break
3. < *kind-of* "break" → media break
4. < *kind-of* "media break" → existing user tasks, existing manual tasks

For the second and third command input, the already calculated result values are returned. Accordingly, the results are: "*break*" and "*media break*". The fourth command requires the analysis of all neighboring *is*-predecessors of the Element *media break*. The Elements *IT system break* and *existing user tasks* have already been analysed based on the DREP used in iKnow (Figure 4). Now the pending analysis of the *is*-predecessor *existing manual tasks* takes place. Since there are two manual tasks in the example process and also one IT system ($r(v_{number\_of\_manual\_tasks}) = 2$ AND $r(v_{number\_of\_IT\_systems}) = 1 \rightarrow$ true), the Knowledge Element *existing manual tasks* receives the return value *true*. The final result of the fourth command input, which particular media breaks exist, is therefore: *existing user tasks* and *existing manual* *tasks*. The following command inputs are used to answer the analysis questions III. and IV., which improvement measures and which aids are suitable:

5. < *used-for* "improvement measure" "handle media break"
6. < *used-for* "tool" "improve coordination"
7. < *used-for* "tool" "improve communication"
8. < *used-for* "tool" "define rules"

The fifth command refers to improvement measures. Commands six to eight are used to identify appropriate aids. The result on the question which improvement measures are suitable for the analysed example process (5.) comprises: *define rules*, *improve communication* and *improve coordination*. In order to determine the appropriate aids for each determined improvement measure, command six results in the following aids: *workflow management system* and *groupware system.* The seventh command contains only the result value *groupware system*. With the eighth command, an empty set as result value resp. the answer *nothing found* in the log output is returned, since the knowledge base modeled for the example of the evaluation does not currently contain aids for *define rules*.

4 Conclusions

In this paper, the previous rough concept [9] of a tool for automated business process analysis was extended, refined, and evaluated using the example of media breaks. For this purpose, the knowledge base on weak points and improvement measures was modified and Data Sources were equipped with defined interfaces in such a way that the relevant data can be extracted from BPMN models. Due to the deduction algorithms and the machine learning components, the business process analysis tool is able to expand continuously with each use case and to continuously improve the analysis results. The analysis of the example process shows that the tool is executable for BPMN models using the constructed prototype and that the analysis questions (chapter 3.2) could be implemented. In consulting projects, the business process analysis tool can initially be used to support automated situation analysis during the consulting phase of problem analysis. Moreover, within the scope of self-service applications, capable consulting clients could analyse their own processes and determine possible improvement measures or aids without personal consultant involvement, as is expected to be relevant in the near future according to a study of the German National Association of Business Consultants (BDU) [21].

Conceptually, the modelling of more particular characteristics with regard to weak points, improvement measures and aiding tools can further specify the analysis of a process and lead to more detailed results. Additionally, the determination of data synchronization and (de-)central data storage can contribute to a more detailed specification of media breaks. Consequently, a media break should also be able to be analyzed under consideration of further weak points, such as system breaks or data redundancies.

https://doi.org/10.30844/wi_2020_w1-fuessl

Moreover, the further development steps of the business process analysis tool include the conception and integration of a monitored learning process for quality assurance and verification of determined analysis results. In this context, it is necessary to improve the prototype technically and to design a user-friendly interface suitable for consultants. Finally, a broad evaluation with consulting companies is necessary in order to validate the analysis process, and to verify and extend the practicability of results.

References

1. Cole, T.: Digitale Transformation. Warum die deutsche Wirtschaft gerade die digitale Zukunft verschläft und was jetzt getan werden muss! Vahlen, München (2015)
2. Downes, L.; Nunes, P. F.: Big Bang Disruption. In: Harvard Business Review, 91, pp. 44–56 (2013)
3. Hamidian, K.; Kraijo, C.: DigITalisierung – Status quo. In: Keuper, F.; Hamidian, K.; Verwaayen, E.; Kalinowski, T.; Kraijo, C. (eds.): Digitalisierung und Innovation, Springer, Wiesbaden (2013)
4. Werth, D.; Greff, T.; Scheer, A.-W.: Consulting 4.0–Die Digitalisierung der Unternehmensberatung. In: HMD Praxis der Wirtschaftsinformatik, 53, pp. 55–70 (2016)
5. Bughin, J.; Catlin, T.; Hiert, M.; Willmott, P.: Why digital strategies fail. In: MCKinsey Quarterly, January 2018 (2018)
6. Nissen, V.: Digital Transformation of the Consulting Industry – Introduction and Overview. In: Nissen, V. (eds.): Digital Transformation of the Consulting Industry - Extending the Traditional Delivery Model, Springer (2018)
7. Nissen, V.; Seifert, H.; Blumenstein, M.: Virtualisierung von Beratungsleistungen. Qualitätsanforderungen, Chancen und Risiken der digitalen Transformation in der Unternehmensberatung aus der Klientenperspektive. In: Deelmann, T.; Ockel, D. M. (eds.): Handbuch der Unternehmensberatung, Berlin (2015)
8. Füßl, F.: Entwicklung eines Modells zur Anwendung inferenzfähiger Ontologien im Software Engineering. Dissertation an der TU Ilmenau, Fak. IA, FG Swarch-Pl (2016)
9. Füßl, A.; Füßl, F.; Nissen, V.; Streitferdt, D.: A Reasoning Based Knowledge Model for Business Process Analysis. In: Nissen, V. (eds.): Digital Transformation of the Consulting Industry - Extending the Traditional Delivery Model, Springer (2018)
10. van der Aalst, W.: Process Mining. Data science in action. Springer, Berlin, Heidelberg, New York, Dordrecht, London (2016)
11. Jochem, R.; Balzert, S.: Prozessmanagement. Strategien, Methoden, Umsetzung. Symposion Publ, Düsseldorf (2010)
12. Peffers, K.; Tuunanen, T.; Rothenberger, M. A.; Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. In: Journal of Management Information Systems, 24, pp. 45–77 (2007)
13. Schreiner, M.; Hess, T.; Benlian, A.: Gestaltungsorientierter Kern oder Tendenz zur Empirie? Zur neueren methodischen Entwicklung der Wirtschaftsinformatik. Arbeitsbericht, Institut WI und Neue Medien, Fak. BWL, LMU München (2015)
14. Krishnamoorthy, C.; Rajeev, S.; Chen, W.: Artificial Intelligence and Expert Systems for Engineers. Boca Raton: CRC Press (2018)
15. Füßl, A.; Nissen, V.; Füßl, F.; Dopf, S.: Entwicklung eines Werkzeugs zur Geschäftsprozessanalyse auf Basis eines inferenzfähigen Wissensmodells. In: Bruhn, M.;

https://doi.org/10.30844/wi_2020_w1-fuessl

Hadwich, K. (eds.): Forum Dienstleistungsmanagement - Automatisierung und Personalisierung von Dienstleistungen, (in press) (2020)
16. Allweyer, T.: BPMN 2.0 - Business Process Model and Notation. Einführung in den Standard für die Geschäftsprozessmodellierung. BOD, Norderstedt (2015)
17. Klettke, M.; Meyer, H.: XML & Datenbanken: Konzepte, Sprachen & Systeme. dpunkt.verlag, Heidelberg (2003)
18. Skulschus, M.; Wiederstein, M.; Winterstone, S.: XSLT, XPath und XQuery. Comelio-Medien, Berlin (2011)
19. Ertel, W.: Grundkurs Künstliche Intelligenz. Eine praxisorientierte Einführung. GWV Fachverlage GmbH, Wiesbaden (2009)
20. Russell, S. J.; Norvig, P.: Artificial Intelligence. A Modern Approach. Pearson, Boston (2010)
21. Nissen, V.; Füßl, A.; Werth, D.; Gugler, K.; Neu, C.: On the Current State of Digital Transformation in the German Market for Business Consulting. In: Nissen, V. (eds.): Advances in consulting research. Recent findings and practical cases, pp. 317–339, Springer International Publishing, Cham (2019)

https://doi.org/10.30844/wi_2020_w1-fuessl