UNIVERSITÄT KOBLENZ · LANDAU

Fachbereich 4: Informatik

Recommender Systems for Process Modeling Tools – A Literature Review

Masterarbeit

Zur Erlangung des Grades Master of Science (M.Sc.)
im Studiengang Wirtschaftsinformatik

vorgelegt von

Tim Johannes Weiskopf

Gutachter:
Prof. Dr. Patrick Delfmann

Koblenz, im Oktober 2019

Table of contents

List of figures

List of tables

Abbreviations

[BN]	Bayesian Network
[BP]	Business Process
[BPM]	Business Process Model
[BPMN]	Business Process Model and Notation
[BPMS]	Business Process Management Systems
[GED]	Graph Edit Distance
[SED]	String Edit Distance

Abstract

To construct a business process model manually is a highly complex and error-prone task which takes a lot of time and deep insights into the organizational structure, its operations and business rules. To improve the output of business analysts dealing with this process, different techniques have been introduced by researchers to support them during construction with helpful recommendations. These supporting recommendation systems vary in their way of what to recommend in the first place as well as their calculations taking place under the hood to recommend the most fitting element to the user. After a broad introduction into the field of business process modeling and its basic recommendation structures, this work will take a closer look at diverse proposals and descriptions published in current literature regarding implementation strategies to effectively and efficiently assist modelers during their business process model creation. A critical analysis of presentations in the selected literature will point out strengths and weaknesses of their approaches, studies and descriptions of those. As a result, the final concept matrix in this work will give a precise and helpful overview about the key features and recommendation methods used and implemented in previous research studies to pinpoint an entry into future works without the downsides already spotted by fellow researchers.

Keywords: Recommender Systems, Business Process Modeling, Literature Review

1 Introduction

The representation of processes in an organization can be visually displayed in a Business Process Model (BPM) or a Business Process Model and Notation (BPMN) which are usually conducted by business process analysts who should be able to capture the organizations business requirements. These models are aiming to provide a better understanding of complex business processes and tasks even for non-business users by using human understandable notations like BPMN. Designing a good and helpful BPM(N) will enable a better understanding of ongoing processes and therefore an easier and productive communication between the business process analysts and the IT experts. (Deng et al., 2017; Kluza, Baran, Bobek, & Nalepa, 2013)

Also, the precise modeling of business processes can be used in current information systems which are thereby able to become process-aware and include allocations of resources, communication services or hardware devices to users. (Hornung, Koschmider, & Oberweis, 2009)

However, the process of creating a proper visual representation of a business process is highly complex, error-prone and takes a lot of time due to extensive knowledge which the analysts require in terms of correct usage of BPM(N) elements and the richness of the modelling language as well as the company's structure, operations, business rules and eventual dependencies. (Deng et al., 2017; Kluza et al., 2013)

Despite these downsides it becomes more and more important for companies and enterprises to make use of this efficient and accurate way of representing their business processes in a model to deal with the fact of growing complexity and fast changing standards in the commercial environment. (Deng et al., 2017; Kluza et al., 2013)

In order to support the business process analysts at their task of designing and creating a proper BPM(N) there are many tools and methodologies already existing on the market especially for using Business Process Model and Notation (Kluza et al., 2013). Nevertheless, these tools do not support the modeler in terms of recommendation mechanisms for the BPMN. There are guidelines, frameworks and style directions for the analysts, yet the process remains a challenging task especially for modeler with less experience.

The usage of recommendation based methods during the BP modeling process can address this challenge and would facilitate it significantly. (Bobek, Baran, Kluza, & Nalepa, 2013)

As empirical studies have proven, users prefer to receive and use suggestions during modeling processes, several approaches to recommendations in BP modeling have been developed. They are based on different factors such as labels of elements, current progress of modeling process or additional pieces of information like process descriptions or annotations. (Bobek et al., 2013; Koschmider, Hornung, & Oberweis, 2011)

The effort for modeling the processes and implementation in information systems could be substantially reduced if existing business process models are able to be reused in an automatic fashion based on the current progress or additional pieces of information during the modeling process. Various features can assist the modeler i.e. by providing autocomplete mechanisms that suggest possible process fragments to be used next or by naming model elements to fulfill business internal standards. Making use of these suggestions can reduce the number of errors during the design process and could potentially speed up the whole process by reusing existing representations of business processes, especially if an existing process repository storing these fragments can be provided. (Hornung et al., 2009; Kluza et al., 2013)

A visual representation of the foundation for recommending process fragments from a repository can be seen in Figure 1. It includes good and bad examples of available process fragments for an edited business process.

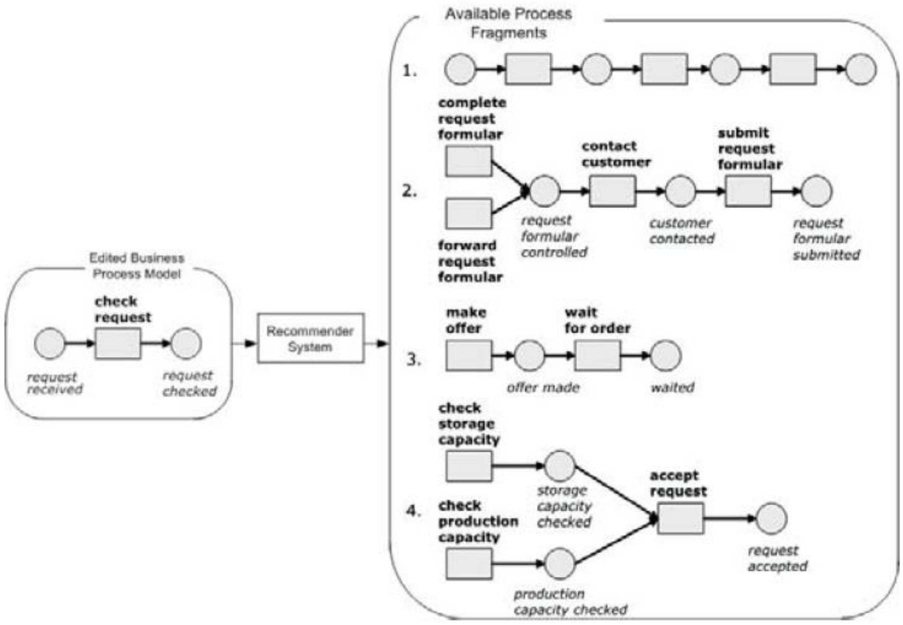

Figure 1- Pool of available process fragments (Hornung et al., 2009, p. 1)

The available process fragments in Figure 1 are analyzed by the recommendation system that has been invoked. In order to provide a helpful recommendation, these should only contain syntactically correct elements and fragments. This criterion of syntactically valid connections to the previous element are violated in the first available process fragment. Thus, this process fragment is one of the bad examples and will not be recommended to the business process modeler.

The second criterion of correctness is compliance of structural properties, which includes, that every AND-split is completed by an AND-join as well as every OR-split is completed by an OR-join. Meaning, that it would violate the structural integrity if a parallel flow initiated by an AND-split would be synchronized by an OR-join. Applying this, would lead the recommender system to not propose the fourth process fragment, since the OR-split is synchronized by an AND-join.

Following these two correctness criterions, a modeler would avoid undesirable deadlocks in the process model and the invoked recommendation system should be able to present only valid fragments to help reduce syntactical errors.

Since recommender systems usually include more than four process fragments, it is necessary, that the recommendations are also based on fitness in respect to the current content and level of abstraction. Here, the developers of recommender systems face the problem of different vocabular used to name the elements in the edited process model fragments and the ones which are stored in the process repository. In order to automatically detect synonyms or homonyms in the business process, it is necessary to implement a semantic similarity measure.

For modeling a business process, it is crucial to maintain a homogeneous level of abstraction in element names and decomposition level. Modeling a process at top level only formulates an overview of the process activities while bottom level models include a detailed description.

Applying this to the example in Figure 1 results in the conclusion, that the second process fragment is contextually correct since it also contains request handling, yet the level of abstraction is very detailed. The third option on the other hand fits the context as well and matches the respective level of abstraction. Only the third option should therefore be recommended to the modeler. (Hornung et al., 2009)

1.1 Objective

The objective of this work is to review the current relevant literature addressing the above-mentioned topic in order to structure, summarize and analyze the current state of research in the area of recommendation-based techniques and autocomplete tools for the creation of process models with a focus on business process models in particular. This work targets existing as well as solved problems regarding the process of development of such tools and to identify possible research gaps.

Furthermore, the aim of this work is to facilitate an entry into the topic itself by describing related topics and to present a structured overview of existing research results, solutions and problems in relevant research contributions in order to find motivation and guidance for developing a tool which provides autocompletion and recommendations in the BP modeling process.

In summary, the following research question arises:

In which area of the current existing relevant literature regarding the topic of recommendation techniques and autocompletion tools for business process modeling are research gaps?

Literature will therefore be analyzed regarding different techniques for recommendations, their influence on quality and user experience as well as pros and cons of them.

1.2 Structure

The structure of the thesis will be made up of the following chapters.

At the beginning in chapter 1 an introduction with motivation and problem definition should inform about the goal of the thesis. This chapter also includes the objective of this work as well as its structure.

To give the reader a basic insight into the area of interest, the most important topics are explained in detail in chapter 2, including an introduction to business process modeling and fundamental knowledge for recommendation system related approaches such as similarity measures or the different bases for recommendations itself.

Chapter 3 presents a precise view about the applied methodology and used search terms. Also, the definition of review scope as well as the literature selection process are further described.

The main part, chapter 4 and 5 of this work, will focus on the evaluation, structuring and classification of the selected literature in relation to the problem and the research question.

In the last part of the thesis, the author presents a retrospective summary of the collected information with details of limitations and problems during the research. Finally, further research needs on this topic are presented and described.

2 Fundamental Knowledge

2.1 A Business Process

A Business Process (BP) is a set of one or more linked procedures or activities that work together in a predefined order achieving a business objective or political goal, usually within the context of an organizational structure that defines functional roles or relationships. A process can be completely contained in a single organization unit or it may include several different organizations. (Chinosi & Trombetta, 2012)

The collaboration of business processes across company boundaries is a complex task due to the lack of clear semantics for the terminology used in their BP models and the use of different standards in BP modeling and execution.

Business Process Management (BPM) offers control over the process environment of an organization to improve agility and operating performance. It is a systematic approach to improve business processes of any organization. BPM is not a specific technology nor is it closely bound to the creation of diagrams or a system architecture. (Chinosi & Trombetta, 2012)

Business process modeling is instead defined as a time period when manual and / or automated (workflow) descriptions of a process are electronically defined and / or modified. As both business process modeling and business process management share the same acronym (BPM), these activities are sometimes confused with the other. Business process modeling is the activity of representing processes of a company so that the current ("as is") process can be analyzed and improved ("to be") in the future (zur Muehlen, 2008).

Business process modeling is typically done by business analysts and managers trying to improve the process efficiency and quality. The term "Business Process Modeling" was introduced around the 1960s in the field of systems engineering. In the 1990s, companies began to replace terms such as "procedures" or "functions" with the terms "processes" and "workflows". (Chinosi & Trombetta, 2012)

Early on there was no tool support for workflows and they were executed from a single person who had to remember all the execution steps of the process. The first work providing aid for the executing users by describing the workflows and documenting the process steps appeared in 1993. Following this, the next change affected workflows in distributing operating cycles among different users instead of being executed by an individual. This development led to the possibility for users to access a user interface connected to the application logic. In 2005 it became possible to change the underlying workflow model or technologies during the users daily routine and without them taking notice, which led to an increase of distance between the business process modeling and the physical execution level. (Chinosi & Trombetta, 2012)

Nowadays research efforts in this area are oriented towards simplifying workflows and their underlying process models so that executioners only see the tasks they have to fulfill. Also, it is beneficial to separate responsibility between different users. They should have their own set of permissions linked to the actions they are performing on the workflow. (Chinosi & Trombetta, 2012)

2.2 Relevance of BPMN

Anyone who wants to manage business processes must describe and document them. There are various possibilities for this. In the simplest case, textual or tabular descriptions are used. Often, presentation or graphics programs are used to create simple flowcharts. They usually consist of boxes and arrows, whereby no particular methodology is followed. (Allweyer, 2015)

This is not enough to accurately represent more complex processes with all relevant aspects, such as branching rules, events, executing organizational units, data flows, and so on. For this purpose, suitable notations are needed. Among other things, a notation for graphic business process modeling determines which symbols are used to represent the various elements of processes, what exactly they mean and how they can be combined with each other. Such a notation is therefore a unified language for describing business processes. (Allweyer, 2015)

Anyone who knows this language will be able to understand the diagrams made by other modelers. A consistent presentation also allows processes to be analyzed systematically or to simulate their dynamic behavior. The increasingly relevant topic of „Governance, Risk and Compliance" (GRC) also requires the establishment and uniform and complete documentation of suitable processes to ensure that all legal and industry-specific requirements about risk management, quality management, security, etc. are met. (Allweyer, 2015)

Finally, models also serve as a basis for the development of information systems to handle and support business processes. Here, too, it must be ensured that the models have a uniform structure and contain all information relevant to system development. (Allweyer, 2015)

Increasingly, business process management systems (BPMS) are used to execute processes. A BPMS contains a process engine that controls processes directly using suitable process models or formal process descriptions. For this, the models must meet very strict requirements, since they are not implemented by people in a computer program but are processed directly by a machine. (Allweyer, 2015)

Over time, various notations for process modeling emerged. Often, these were proprietary notations of specialized modeling tools or workflow management systems. Standards have been developed in the field of BPMS executable process descriptions, such as: XML Process Definition Language (XPDL) created by the Workflow Management Coalition ("Workflow Management Coalition," 2019) and BPEL (Business Process Execution Language) as an Oasis Standard ("Oasis," 2019).

XPDL and BPEL, as text-based XML descriptions do not include graphs and are limited to the definition of automatically executable processes. In the field of business process modeling, the notation of the event-driven process chain (EPC) is often used, which was widely disseminated before the development of the BPMN standard. However, this is not a standard, and in the meantime, EPC is being replaced by BPMN in many places.

Most EPC modeling tools today also allow process modeling with BPMN. Other standards, for example, the activity diagrams of the Unified Modeling Language (UML) have not been able to prevail in practice for business process modeling. Their use has been largely limited to the area of object-oriented software design, where UML is the accepted standard.

Yet, there are many different modeling methods still available which each have a different focus (data-oriented, object-oriented or process-oriented). Recker, Muehlen, Siau, Erickson, & Indulska, 2009 summarized a few of the most popular ones in use today in the following Table 1.

Table 1 - Popular modeling methods (Recker et al., 2009, p. 3)

Method	Focus
Data flow modeling	Describing the flow of data structures in partitioned systems
Entity-relationship modeling	Describing structures of data bases on a conceptual level
Enterprise modeling	Describing and providing a graphical overview of the structure of organizations
Unified modeling	Describing systems in the form of encapsulated objects
Process modeling	Describing business operations and the dynamics and behavior of information systems
Object-process modeling	Describing both the structural and the dynamic aspects of a system via the building blocks object and process

Business Process Modeling (BPMN) has become widely accepted as the standard for process modeling in recent years. BPMN is the process modeling notation that most modeling tools support, as market analysis shows (Lübbe, Allweyer, & Schnägelberger, 2015). According to Recker from 2010, BPMN is supported by more than 60 commercial and academic process modeling products and is finding rapid adoption in industry (Recker, 2010).

The website *www.bpmn.org* also has a list of over 60 implementing tools that support BPMN modeling ("bpmn.org," 2019). Many organizations train their process management staff in BPMN modeling and roll out BPMN as an enterprise-wide modeling standard. In a recent survey among modeling tool users, BPMN was the most commonly used process modeling notation (Lübbe et al., 2015). (Allweyer, 2015)

While BPMN has found widespread adoption as a graphical notation, it was developed without a formal underlying metamodel, and did not provide a specification for a persistency format such as an XML schema or a file format. The BPMN 1.1 and 1.2 specifications provide a description of the BPMN modeling constructs using class-diagrams, but these diagrams do not express restrictions that govern the use of symbols (Recker et al., 2009). The Development and problem solving of this will be explained in detail in the next chapter.

2.3 Development of the BPMN

BPMN was originally developed by the Business Process Management Initiative (BPMI), a consortium consisting mainly of representatives of software companies, beginning their efforts in October 2001 (Recker et al., 2009). The initial purpose was to provide a graphical notation to represent process descriptions of BPML (Business Process Modeling Language). BPML was used to specify process descriptions that can be executed by a BPMS, similar to BPEL. The development of the BPML is no longer continued, it was abandoned in favor of the more popular BPEL. (Recker et al., 2009)

A first draft of BPMN was publicly released in August 2003 but the work on the specification continued. In June 2005, the Business Process Management Initiative (BPMI) merged with the Business Enterprise Integration Domain Task Force of the OMG. A revised BPMN specification was sent to a finalization process within the OMG and published as an official OMG specification in February 2006. Updated specifications of the notation have been released in February 2008 (version 1.1 with extensions to the event symbols and a new signal event type) and January 2009 (version 1.2 with editorial changes). After rather small enhancements in versions 1.1 and 1.2, the 2011 version 2.0 brought extensive changes and enhancements. (Recker et al., 2009)

The BPMN 2.0 specification extends the scope and capabilities of the BPMN 1.2 in several areas: it formalizes the execution semantics for all BPMN elements, defines an extensibility mechanism for both process model extensions and graphical extensions, refines event composition and correlation, extends the definition of human interactions, defines choreography and conversation models (a mean for better modeling interactions), and also resolves known BPMN 1.2 inconsistencies and ambiguities. Furthermore, BPMN 2.0 defines a meta-model and a diagram definition model along with accompanying interchange formats both XMI and XSD based. (Chinosi & Trombetta, 2012)

A wide collection of new constructs is available in BPMN 2.0 given its duality with respect to create diagrams to communicate or modeling for execution (e.g., there are 336 possible depiction permutations just for tasks). With BPMN it is now possible to model a different set of processes, such as Orchestrations (both private non-executable and private executable (internal) Business Processes, Public

Processes), Choreographies and Collaborations, which can include Processes and/or Choreographies and a view of Conversations. Also, data acquired a greater importance in BPMN 2.0. In fact, data is no more part of the Artifacts but is a separate element category, including Data Input/Output, Collection Data Objects, Data Store and Messages. (Chinosi & Trombetta, 2012)

The most recent version of the specification document from 2014 bears the number 2.0.2 (Object Management Group, 2014). It contains only minimal corrections to the text, but no content changes compared to BPMN 2.0. Since 2013, BPMN has also been an official ISO standard (International Organization for Standardization/International Electrotechnical Commission & others, 2013). The latest version of the BPMN specification can be found publicly available at www.omg.org/spec/BPMN (Allweyer, 2015).

To visualize the development of the above-mentioned standards, Figure 2 organizes them in a timeline from 1994 till 2011 when XPDL 3.0, BPMN 2.0 and BPEL4People became a standard.

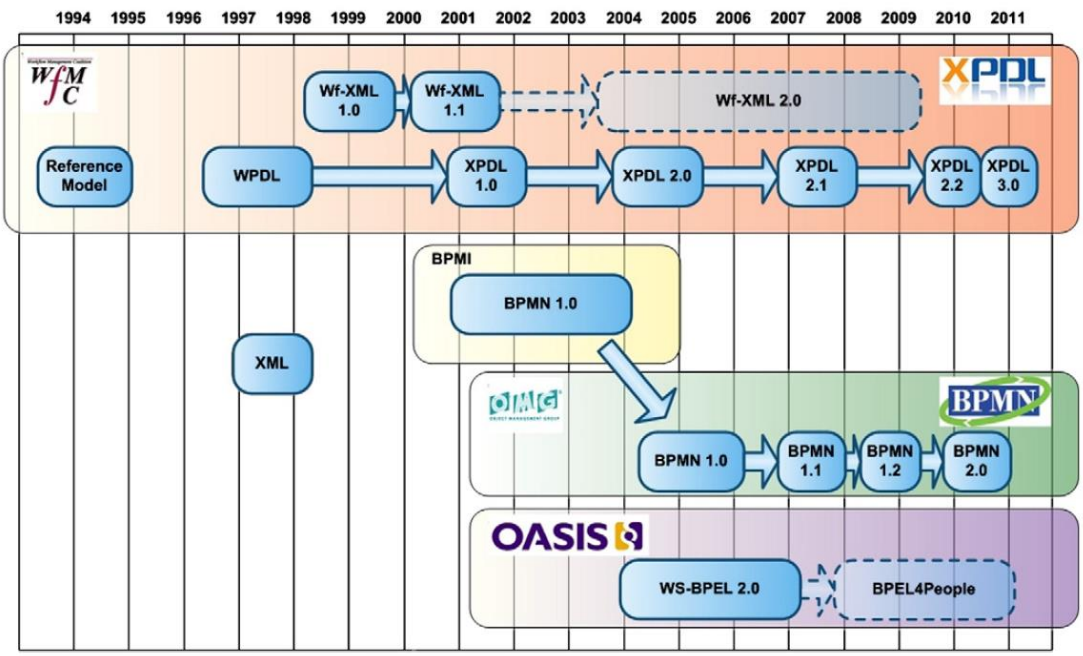

Figure 2 - Business processes related standards time-line (Chinosi & Trombetta, 2012, p. 128)

2.4 Content of BPMN 2.0

For most BPMN users, the graphical representation of the processes should be the most important. For this, BPMN offers a total of three diagram types:

1. Process or collaboration diagram: This allows the process flow with the individual activities, branches, etc. In addition, collaborations of two or more processes can be modeled. The interaction of the processes takes place via exchanged messages. The process and collaboration diagram is the same type of chart. A single-process diagram is commonly referred to as a process diagram, one with multiple interacting processes as a collaboration diagram. (Allweyer, 2015)

2. Choreography diagram: Like collaborations, this is about the exchange of messages between different partners. However, it is no longer the individual processes of the partners involved that are modeled, but only their interaction. Each message exchange is displayed as a separate activity and you can also change the order of the messages exchange branches, loops and others to model more complex exchange protocols between processes. (Allweyer, 2015)

3. Conversation diagram: This is a presentation of several partners with their communication relationships. (Allweyer, 2015)

Most commonly, process or collaboration diagrams are used. In part, BPMN tools and books are limited to this type of diagram. While undoubtedly most important, there are also potential uses for the other two types of diagrams, which is why the author mentions them as well. The BPMN specification not only verbally explains the various graphical notation elements and rules for modeling, but also defines them in the form of a meta-model. UML class diagrams describe the various BPMN constructs and their relationships to each other. Such a meta-model is more accurate and clearer than purely verbal descriptions. In addition, the meta-model contains additional language constructs that are not displayed in the graphical models. For example, they are needed by process engines to capture additional information needed for process execution. As an ordinary modeler, you do not have to deal with the metamodel. In general, you will use modeling software that ensures that you can only create models that match the specification and therefore the meta-model. The main addressees of the meta-model are thus rather the manufacturers of modeling tools, process engines and similar software. (Allweyer, 2015)

The meta-model is also the basis of an exchange format for BPMN models. Previously, with few exceptions, it was not possible to transfer the BPMN models created in one tool to another tool. Since version 2.0 a standardized exchange format is available. Several tool manufacturers support this standard format, allowing BPMN models to be used between different modeling tools, as well as an exchange between a modeling tool and a BPMS. In practice, however, the implementations of this exchange format are not yet completely uniform, so that when exchanging models occasionally certain difficulties or losses of some details may occur. (Allweyer, 2015)

If you want to automate BPMN-modeled processes using a process engine, you must specify how the various modeling constructs should be executed. This execution semantics is also specified in the specification. This is to ensure that one and the same model is interpreted by different process engines the same way and thus executed in the same way. Also, the BPMN execution semantics have not been implemented completely uniformly by all BPMS manufacturers, so that there may also be some differences in the execution of the same model on different process engines. However, despite the deviations that sometimes occur, both the exchange format and the execution semantics are very useful, as otherwise no inter-manufacturer model exchange would be possible and significantly greater differences would occur in the process execution. In the first version, the acronym BPMN still stood for "Business Process Modeling Notation". With version 2.0 this has been changed to "Business Process Model and Notation". This expresses that BPMN includes not only graphical notation, but also the metamodel, interchange format, and execution semantics. (Allweyer, 2015)

2.5 Professional and Executable Models

The BPMN was originally developed in the context of process descriptions that can be executed by the Process Engine of a workflow or Business Process Management System (BPMS). However, the developers of the BPMN have the claim that this notation can be used to create both technical and subject-oriented models. BPMN is intended to be a common language of business experts and IT experts.

And indeed, BPMN is used for both purely process modeling and near-execution modeling. For example, the modeling tools listed in (Lübbe et al., 2015), which for the most part offer BPMN as a process modeling notation, are predominantly subject-specific tools.

Despite the use of a common notation, subject specific and technical models differ very clearly in practice. For professional models, the understanding of the basic process flow is in the foreground. Therefore, it is omitted to represent too many details. For example, conditions at branches are formulated in plain text rather than in the form of exact logical expressions. Exceptions and rare cases are often not modeled but explained with the help of annotations and descriptions. (Allweyer, 2015)

The origin of some BPMN constructs is clearly in the range of executable process specifications. Therefore, the BPMN language inherits some special loop constructs, exception handling, and transactions.

Programmers and IT professionals are familiar with these topics. In technical process models, such things are usually not found. (Allweyer, 2015)

Accordingly, only a part of the entire notation is used in the technical modeling. Some BPMN experts believe that some of these more engineering-related constructs should also be used for professional modeling, so that from a professional point of view relevant exceptions and their treatment in the process can be properly displayed.

Allweyer points to the well-known 80-20 rule. In his estimation, 80% of the costs, delays and errors are caused by 20% of all cases, namely the exceptions. Examples include cancellations, out-of-commerce items or missed time limits. Anyone wishing to use the BPMN for functional process modeling should therefore decide in advance which constructs to use and how certain facts should be presented. It makes sense to commit such decisions in the form of modeling conventions. If the processes modeled at the technical level are subsequently to be automated by a process engine, then it is still necessary to specify how the technical models are to be converted into execution-oriented models, which includes how necessary additions, restructurings and detailing are carried out. (Allweyer, 2015)

2.6 Problems of BPMN 2.0

After introducing the language for modeling business processes in detail, it is worth mentioning, that the OMG standard for BPMN 2.0 does have its problems, as Börger describes in his work 2012.

It outlines, that the usage of BPMN 2.0 could not guarantee platform-independent and consistent communication between different stakeholders which should be a main objective of a standard. The author claims, that the descriptions and specifications of relevant concepts contain ambiguities and therefore leave space for different interpretations in design, analysis and use of BPs. As an example for this underspecification, the lifecycle concept and the relating interruption mechanisms are brought up. It is not clearly specified how they interact with the process in a nested scope. Another concern is mentioned for evaluating expressions. The standard lacks information about the timing of evaluating a modeled expression. This could happen either before or at process start, upon state change or when a token becomes available. It gets even more complicated in the case of multiple events using the same expression definition. (Börger, 2012)

Further, the author states, that the data management of an executable BPMN is compiler-dependent and therefore not portable. Data objects, which are connected to activities, are used only informally instead of referring to a shared location. Another shortcoming of the standard is that resources can only be managed via lanes or performers of tasks and there is no support for good process structure which could lead the modeler to create complex and confusing diagrams. (Börger, 2012)

The mechanism for refinement from conceptual into an executable BPMN is not clearly provided by the standard document, which would be necessary to guarantee reliable implementations of these especially for platform-independent ones. Above that, executable models are hardly backwards compatible to be analyzed by human readers. (Börger, 2012)

In conclusion, these underspecifications and ambiguities in the standard document could lead to different interpretations by different stakeholders which would make the communication between them unreliable instead of easier. Also, the model could be executed differently by different compilers. These weaknesses should be taken into account when BPMN is used for modeling. (Börger, 2012)

2.7 Semantic similarity

In E-Business, enterprises collaborate across organizational boundaries to perform common tasks. But, even when sharing similar demands, enterprises are using their specific vocabulary and structural representations for modeling business processes. By using formal languages such as Petri nets for modeling business processes, purely syntactic composition problems of interorganizational business environments may be solved. (Ehrig, Koschmider, & Oberweis, 2007)

However, a missing semantic representation of Petri net elements can hamper further interconnectivity of business processes. When enterprises decide to interconnect business processes, synonyms, homonyms or similar labeled process elements have to be identified to avoid misunderstandings. Furthermore, in order to understand business processes, significant experience in the field of business process engineering and effort to check differences between the respective business processes are required. By describing business process models in an unambiguous format which enables computer reasoning, the automation of process composition can be facilitated. These so-called semantic business process models promise appropriate business process discovery, interoperability and interconnectivity. (Ehrig et al., 2007)

The automation of process discovery can help to accelerate finding appropriate composable business process models faster than manually discovering business process models. Providing a (semi-)automatic approach for process interoperability and interconnectivity helps to save costs and time when establishing interorganizational business collaborations. (Ehrig et al., 2007)

2.8 Level of Abstraction

The level of process decomposition depends on the modeler and on the purpose and scope of the process modeling. Each decomposition level describes process elements from a different abstraction level. Top level process models formulate an overview of process activities and bottom models provide more detailed descriptions. However, users have to maintain particular modeling requirements such as homogenous abstraction of process element names on the same decomposition level in order to improve model consistency. But model consistency demands to identify an appropriate point to stop the decomposition and to avoid overly granularity for process element names (e.g., datatypes for process element names). Consequently, to maintain this particular modeling requirement demands a significant amount of experience in the field of process engineering and may result in extra analyzing efforts. (Koschmider & Blanchard, 2007)

2.9 Subject based Recommendations

This kind of process recommendation techniques can assist modeling new processes by recommendations according to specific conditions including fitness, syntactical correctness, and structural correctness.

The suggestion itself is not directly dependent on the context it is currently placed in, it may actually inspect the existing context in order to deliver a more accurate result of it instead of recommending new items. These recommendations could include attachments for the business process element already modeled, like attaching an external entity to it. This recommendation appears naturally where the user should link two already existing elements, for example a decision table attached to a gate to describe its conditions, a catch event for a throwing intermediate link event, a service task performed in the given task or a subprocess that should be linked with a given activity. Furthermore, subject based recommendation includes structural recommendations where a new part of the diagram is suggested to satisfy its structural correctness. This includes recommendations for missing incoming or outgoing flows, missing joins and more. Also included are textual recommendations for naming an element based on the current typing of the user or even of the current context of the model. (Kluza et al., 2013)

Subject-based fragments (nodes) recommendation plays an efficient role only if the context of new businesses can be obtained directly or inferred, which turns to be a challenge to carry out this kind of recommendations. Therefore, when lacking of contextual information, researchers seek the position-based recommendation. (Deng et al., 2017)

2.10 Position based Recommendation

This kind of position-based methods can recommend fragments (nodes) starting from the position where the process is being edited according to the relationships between process fragments and activity nodes extracted from existing process stored in a repository. This could be performed in different directions, such as forward or backward completion. Forward completion includes, that a part of the process is known and the rest of it, starting at the selected activity, is to be suggested as displayed in Figure 3. Backward completion suggests process fragments which end at the selected activity displayed in Figure 4. In both figures the displayed Tasks 1 to 5 are the ones suggested and they connect to the selected activity with the red flow connector. (Kluza et al., 2013)

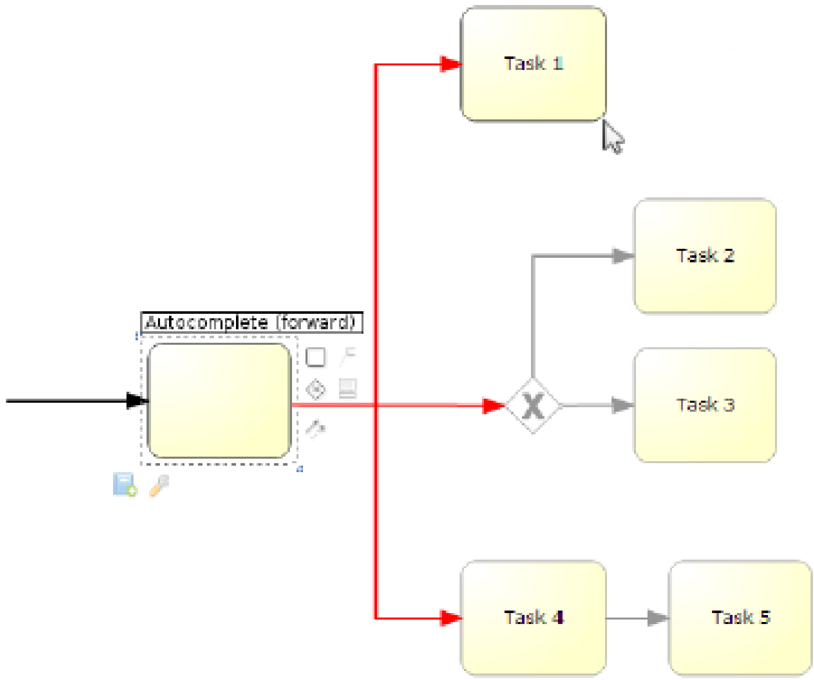

Figure 3 - Forward completion (Kluza et al., 2013, p. 51)

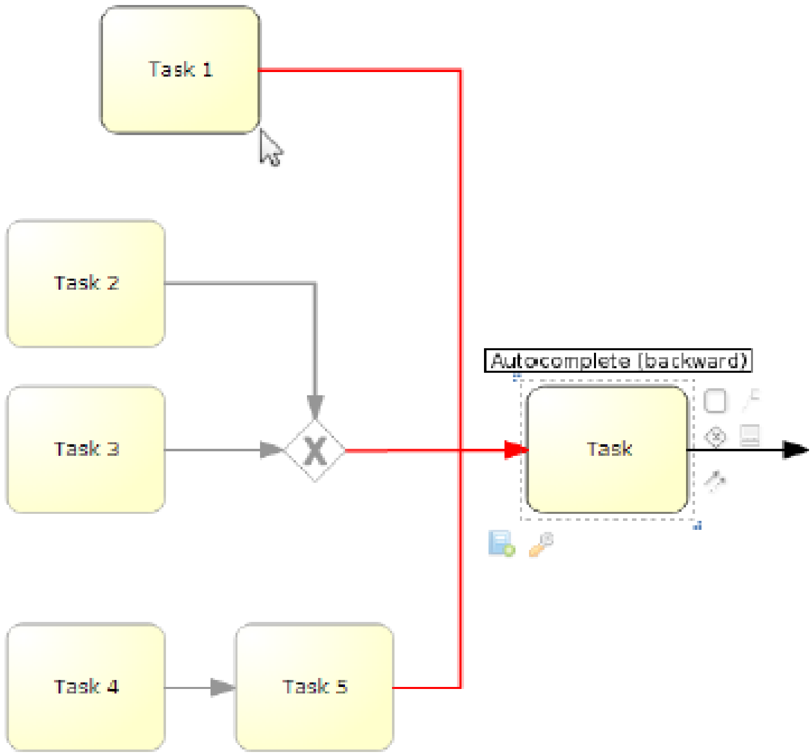

Figure 4 - Backward completion (Kluza et al., 2013, p. 51)

3 Methodology

3.1 Structured Literature review

As already mentioned in the structure of the work, the next section will include the evaluation of the literature search.

A review of previous relevant literature is an essential part of any academic project. An effective review provides the reader with a good foundation for expanding his knowledge. It facilitates the development of theories and opens up areas where there is already an abundance of research and identifies areas where research is needed and knowledge gaps exist. The goal of a literature review is also to motivate researchers to fill these gaps. Therefore, the analysis will be followed by a listing for future research. (Webster & Watson, 2002)

Webster and Watson (2002) emphasize that tables and figures are an effective tool to convey knowledge and insights to the reader. By way of example, the authors introduce a table (Table 2) in which articles are incorporated according to the authors' persecuted concepts.

Table 2 - Concept Matrix Example (Webster & Watson, 2002, p. xvii)

Table 2. Concept Matrix

Articles	Concepts				
	A	B	C	D	...
1		✖	✖		✖
2	✖	✖			
...			✖	✖	

Based on this table, found sources are classified according to their key features and recommendation methods. As a further structuring, the analyzed sources are split into their different recommendation bases and ordered by their date of release.

3.2 Framework

The review follows the guidelines and framework proposed by Vom Brocke et al. 2009 which is displayed in Figure 5.

Following this framework, the process of this work is split into five phases. First is the definition of review scope which is described in detail in chapter 3.3. The conceptualization of the topic led to chapters written above about the initial problem as well as fundamental knowledge. Literature search and collection is combined in chapter 3.3 and 3.4. Following this, the main part of this work is the literature analysis and evaluation finishing with chapter 6 including the conclusion and research perspective.

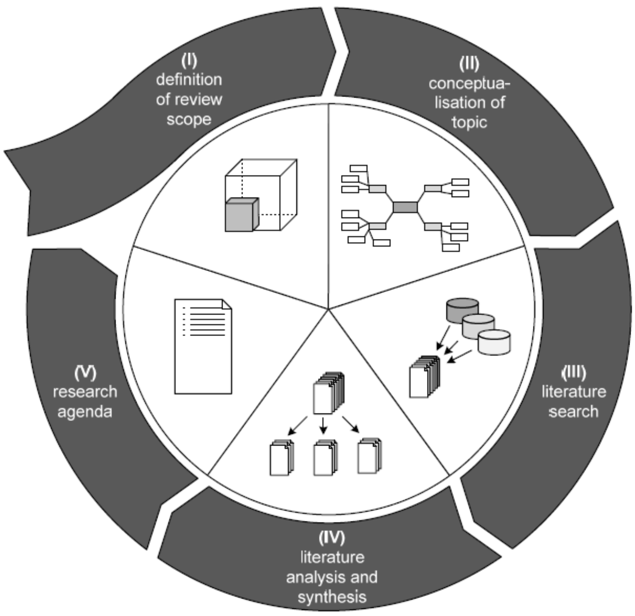

Figure 5 - Framework for literature reviewing (Vom Brocke et al., 2009, p. 9)

3.3 Definition of review scope

In order to achieve the goal of the thesis and to be able to answer the research question, a structured literature search is carried out to identify research papers and articles that describe and / or analyze the current state of the art recommendation systems for business process modeling tools in the literature and in practice as defined above.

According to Vom Brocke et al. 2009 it is necessary to define a research scope and provide a conceptualization of the topic before conducting a literature review.

For the search, an established research tool is queried with specified search words and phrases for literature which was published after the year 2009. After initial research, it turned out that the authors do not always mention the word "system" in relation to recommendation. Often, the term "technique" or "method" is also used. To cover all these possibilities, the first search phrase "recommendations" without further specification is determined. The second phrase is "(business) process modeling" in order to distinguish it from the term "business process", as this often does not refer to the desired literature. These two search phrases are searched for in title, abstract or keywords of the articles.

The following keyword sets were identified based on the definition of the topic.

1. recommendations, recommender systems, recommendation system(s), recommendation technique(s), recommendation method(s), autocompletion, autocomplete, autocomplete system(s)
2. business process modeling, business process model(s), process modeling, process model(s)

For the literature search it is also important, that the results were written in English, be published as a journal article, book, book chapter or proceedings paper and published between the year 2009 and 2018.

The queried research tool is Web of Science.

Additional filtering rules were applied, which refined the results to only cover papers published in the subject of Computer Science.

These filters and conditions combined led to this final search string:

(TS=((recommend* OR autocomplet*) AND ("process model*"))) *AND* **LANGUAGE:** (English) *AND* **DOCUMENT TYPES:** (Article OR Book OR Book Chapter OR Proceedings Paper)

Refined by CATEGORIES: (COMPUTER SCIENCE INFORMATION SYSTEMS OR COMPUTER SCIENCE SOFTWARE ENGINEERING OR COMPUTER SCIENCE ARTIFICIAL INTELLIGENCE OR COMPUTER SCIENCE CYBERNETICS OR COMPUTER SCIENCE THEORY METHODS)

Timespan: 2009-2018. **Indexes:** SCI-EXPANDED, SSCI, A&HCI, ESCI.

For the subsequent literature analysis and structuring, preference is given to sources that have been published in specialist journals of high scientific quality and subjected to an examination by independent experts.

The selected sources are assigned to the different areas of recommendation techniques and autocomplete tools for business process modeling. This mapping is based on keywords used within the text. An attempt is made to conclude these areas even if the areas are not explicitly mentioned by the tasks described. If a system is described that covers several areas of this topic, it is placed in the area that has more relevance and similarities, which is indicated in the description. If in the literature the use is explicitly separated into the different areas, the research contribution is divided into several areas.

In the individual areas, the identified contributions are then analyzed individually. The main statement as well as the characteristics of the introduced system are to be clarified thereby. If the author mentions problems and suggestions for future research they will also be included in the analysis, but will be described in more detail later in the work.

Finally, all areas of the recommendation techniques and autocomplete tools in the classified articles are compared to emphasize the special features of the individual areas and to compare the research contributions.

Querying the Web of Science database with this search string yielded 50 relevant papers. The following visualization (Figure 6) represents the distribution to the different subjects.

Figure 6 - Distribution of subjects

3.4 Literature Selection

In the literature selection phase, all relevant papers abstracts were analyzed regarding their fitness in the desired scope. A large number of these papers study for example business process modeling, yet they just propose recommendations for future research, which fits the search string, but not the topic of this work. Therefore, these papers were excluded for the following literature analysis.

Further, the relevant papers were used to perform a forward- and backward search to complete the list of relevant literature.

The selected literature finally consisted of nine studies which fit the review scope to a desired threshold.

4 Analysis and Evaluation

In this section the selected literature will be profoundly analyzed. Therefore, the author presents at first a summary of the work including the key features of the described system as well as the applied recommendation methods. Based on this summary the author provides an evaluation where the strengths and weaknesses of the work are highlighted, possible missing descriptions or research gaps of their proposals are stressed as well as an outlook for future research is given. The analyzed and evaluated literature is ordered ascending by their release date in order to make proper use of eventual links to previous works. In case multiple authors published their work in the same year, it is then ordered the authors names alphabetically.

4.1 Position-Based Recommendations

This section of the analysis concerns recommendation systems which target a position-based method. This method, as already introduced in chapter 2.10, recommends fragments or nodes based on the position where the process is currently being edited. Key factor for selecting a fitting element from the process repository to recommend to the user is therefore the relationship between already modeled elements in the business process. This might include forward or backward oriented modeling approaches.

T. Hornung, A. Koschmider & A. Oberweis

A Recommender System for Business Process Models

2009

Summary

Hornung et al. 2009 describe a recommender system and explain how to infer correct and fitting process fragments during the business process model editing. The authors outline the required features for such a recommendation system which fits the application scenario in their work.

The authors identified the following key features as requirements for their recommendation system.

The first is transparency, which restricts the number of process elements in one process model diagram. The appropriate number they came up with is 15 elements which is the result of practical tests from different projects the authors observed and analyzed. This amount of elements should be considered in the recommendations from the system. If the user has already modeled five elements, the invoked system should not exceed proposals containing more than ten further elements. Also, part of the transparency requirement is, that stored model fragments should be composed of at least five and maximal ten elements to be beneficial for the modeler. These amounts of process fragments resulted from an interview the authors made with a group of 55 persons across different modeling experiences where none of the modelers selected a proposed fragment with four or less elements during the editing process.

Next up comes flexibility as a required feature for the recommendations which regards the proposing of fitting fragments for different levels of user experience in modeling. The reason for required flexibility is, that expert users can faster and easier understand larger proposed process fragments while beginners need more time and therefore, should get process fragments with lesser elements than experts. Furthermore, the process fragments stored in the repository for proposing should be annotated with the user type who created it to allow reliable assertions about the quality of the fragment.

The modeling technique is another key requirement and should allow the modeler to start forward-oriented as well as backward-oriented. The recommendation system has to take this into account for reasonable proposals during the modeling process left-to-right and vice versa.

To fulfill the last requirement, the stored process fragments in the repository needs to be analyzed before they are proposed during a modeling process to guarantee, that the fragments satisfy syntactical correctness criteria.

In order to support (semi-)automatic recommendations of correct and fitting process fragments the authors describe Petri nets with the Web Ontology Language OWL. The OWL-based description of petri nets enables the described recommender system to handle and manipulate business process models. In order to incorporate the semantic business process models into the recommender system, the authors use a rule language that is capable of reasoning over OWL classes and individuals. They chose the Semantic Web Rule Language (SWRL) as suitable format for their project. With the help of these two languages they came up with a set of OWL properties to define their process fragments and SWRL rules to infer correct and fitting process fragments for an edited business process model based on reasoning techniques. Further, the authors make use of an algorithm presented by Koschmider & Blanchard, 2007 to compare the level of abstraction and propose fragments with a similar level of abstraction or advice refinement in case of a large difference in this level.

The last part for proposing fitting process elements is to verify semantic similarity. The authors make assertions about the similarity between process models by aggregating the calculated degrees of similarity of four measuring techniques proposed and evaluated by Ehrig, Koschmider, & Oberweis, 2007. Due to the combined degree of the similarity measures, fitting process fragments can be detected and proposed for recommendation even when a different vocabular for the same process is used.

The authors finish their work by stating, that the simplicity of their approach makes it possible to apply the proposed recommender system also to other process modeling languages such as BPEL or EPC. Yet, they make remark, that more research work is required in order to store process fragments in different modeling languages and to recommend process fragments to modelers in the currently used modeling language.

Evaluation

The above summarized proposal by Hornung et al. 2009 for a recommendation system based on petri nets, the Web Ontology Language OWL and Sematic Web Rule Language SWRL addresses many desirable outcomes for such a system. Despite the fact, that the authors describe the required key features very detailed, they miss the opportunity to clarify some important applied techniques to fulfill them. They emphasize the importance of recognizing semantic similarity between the edited business process and the stored fragments, yet they do not present on their own which similarity measures they apply. The same applies to the technique or algorithm they use to detect the level of abstraction of process fragments. Yet, they describe how and why they formalized certain SWRL predicates or OWL properties in a detailed and clear way.

After representing the implementation of their recommender system, the authors do not clarify their reached level of success rate in any way. For the reader it is not clear, if the system was able to propose useful and correct fragments from the process repository. The approach from the authors could be picked up and analyzed in future research regarding these downsides to validate their findings.

J. Zhang, Q. Liu, K. Xu

FlowRecommender: A Workflow Recommendation Technique for Process Provenance

2009

Summary

Zhang et al. 2009 present in their work a detailed overview about their research towards a workflow recommendation system called FlowRecommender. The system generally features two main modules which involve pattern extraction and registration from a workflow repository into a pattern table and the workflow recommendations for the user editing a workflow.

During the pattern extraction and registration phase, which is performed offline, candidate nodes are identified by its influencing upstream sub-path that determines their occurrence. The discovered patterns which provide a sufficiently strong correlation between these two are then registered in the pattern table to be ready for recommendation during the modeling process. The confidence for such a node is measured by the probability that this node appears depending on the appearance of its given sub-path. To be identified as a pattern the confidence has to be above a given threshold. The pattern table then stores them in a table connecting the node with its influencing upstream sub-path represented as an ordered sequence of nodes which can be used for comparison by the recommendation module.

While the user interacts with the workflow construction system, the recommendation generating module tries to match the influencing upstream sub-paths stored and registered in the pattern repository against the current workflow to present possible candidate nodes to the user. The workflow under construction thereby serves as the input for the recommendation module. Recommendations by FlowRecommender are offered in a stepwise fashion, meaning, that the system recommends only the next most likely node to the user.

This process will cycle through until the workflow has been constructed to the point such that the desired task has been fulfilled.

In order to proof their proposed system, the authors carried out an effectiveness as well as an efficiency study whose results are presented in their work.

The effectiveness study validated the accuracy of the recommendations provided by their system. Therefore, they set up a test set of a small fraction of workflows to evaluate the outcome of recommendations against the real execution. The accuracy is based on the consideration if the recommended top three nodes contain the desired node. The outcome of the study is compared to two different recommendation methods which are based on other parts of the nodes sub-path. The results of their study outline that FlowRecommender is superior to the compared methods.

In the efficiency study the authors observe, that while the execution time during the pattern extraction and registration phase rises in a linear fashion with the number of workflows processed, the efficiency of recommendations remains independent of the number of registered patterns. The authors outcome of their efficiency study regarding the execution time of pattern extraction for an increasing number of workflows displayed in Figure 7. Figure 8 shows the result of the execution time for recommendations for increasing workflows in provenance.

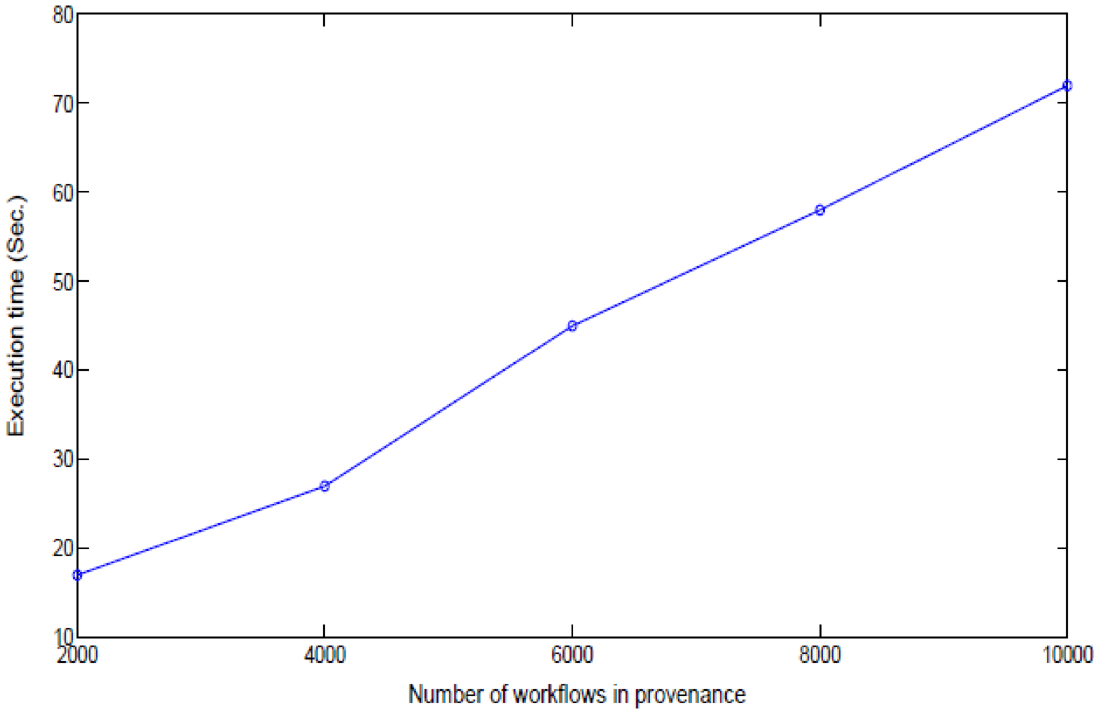

Figure 7 - The efficiency of pattern extraction (Zhang et al., 2009, p. 60)

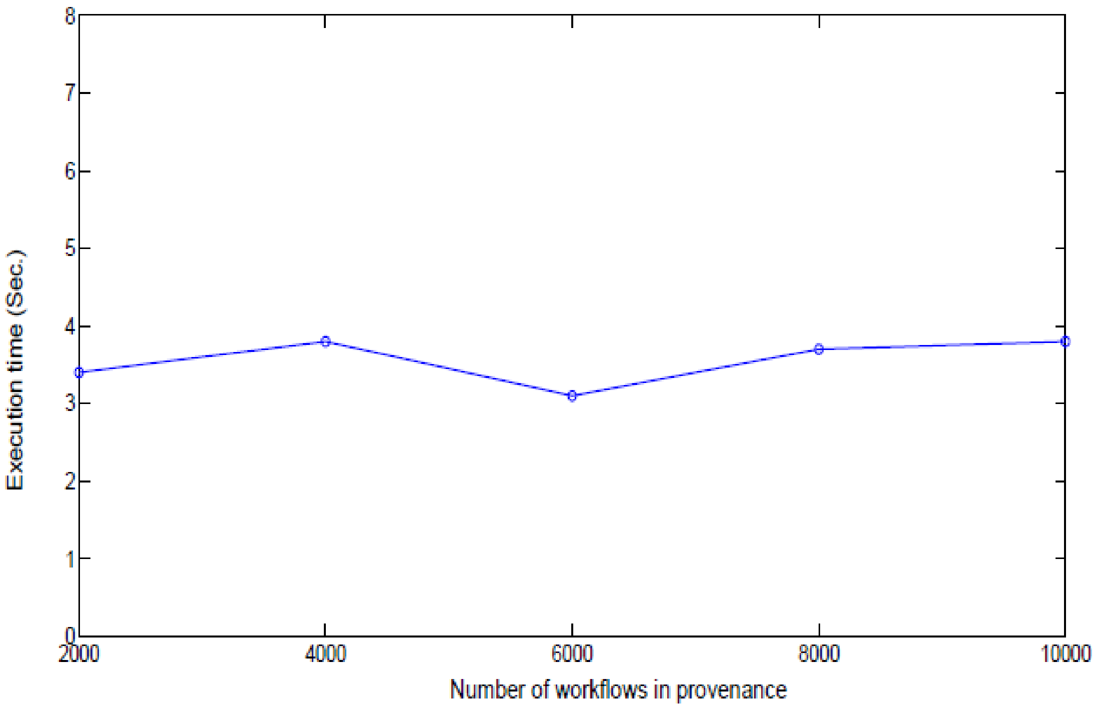

Figure 8 - The efficiency of recommendations (Zhang et al., 2009, p. 60)

There are some further research directions the authors are interested in exploring, including:

1. First, the system is currently only able to register candidate nodes for the most adjacent influencing upstream. However, the authors suggest that there probably exist multiple upstream sub-paths for nodes, which the system is not able to find yet.

2. Second, the authors are interested in researching possible performance boosts for the system by indexing the stored sequence patterns.

3. Finally, the experimental evaluation of the system is only based upon synthetic workflows. The authors plan to utilize another workflow construction system in order to collect real-life workflows for further performance validations of their system. (Zhang et al. 2009)

Evaluation

The above summarized system by Zhang et al. from 2009 called FlowRecommender processes previous workflows to provide recommendations for the best next node that can be chosen to complete the workflow currently under construction. The recommendation method is based upon the influencing upstream sub-path of nodes which calculates the probability of the occurrence for the next node based on the previously modeled nodes. These sub-paths are processed and stored in a pattern table to provide a better performance for the recommendation module. This technique enables the system to provide a steady performance independently of the amount of previously processed workflows. The authors provide a detailed insight into their measuring calculations for the recommendation module. Yet, the processed workflows are based upon synthetic data sets which are not representative for real world applications. Also, the authors do not provide insight into the complexity of their synthetic workflow data sets which could be a drawback and make the system unable to be used with real world data sets. For future research this system could be analyzed and possibly extended to work with real world data sets containing complex structures.

A. Koschmider, M. Song, H. Reijers

Social software for business process modeling

2010

Summary

Koschmider, Song, & Reijers, 2010 propose a recommendation-based process modeling support system that implements two types of modeling support in order to achieve the user's modeling intentions. The first type is a query interface which allows the users to request complete process models or parts of it via logical complex combinations of query parameters. The second type is a recommender component which can be invoked during the users modeling process. The recommendations are then provided based on the selected element group of the business process diagram.

To enable the query interface to search for models or model parts, they need to be indexed. Process model parts are additionally linked to their origin process model. The stored processes can be enriched by seven query arguments which are

- Title: referring to names of process elements (e.g. approved request)
- First Element: searching for a specific first element(s) in the process model
- Last Element: searching for a specific last element(s) in the process model
- Property: referring to specific properties of a process model assigned by users before storing the process in the repository (e.g., standard signifies a standard process)
- Purpose: referring to models fulfilling one of the four modeling purposes such as analysis, documentation, execution or reengineering
- Objective Description: searching for processes fulfilling an objective (e.g., processes modeling handling of order request). This field is only searchable if process builders have annotated the process with the corresponding data before storing the model in the repository
- Previous User Selection: searching for a specific user who selected a recommendation

Furthermore, the authors use a free English taxonomy to overcome the limitation through a controlled vocabulary. The query interface then enables the user to use standard Boolean operators, such as AND, OR, and NOT to express complex queries whose result is displayed in a ranked table. The greatest influence on the ranking is hereby the matching score of the query and the indexed process model parameters. Due to the link between process model parts and the original process model it is part of, the user can preview all eventually related process model parts before and after the queried one. The history of a user selecting a proposed recommendation enables the system to create a social network-based recommendation support. These social networks can be based on the strength between organizational units, similarity between different users or the insertion history of the users, see Table 3.

Table 3 - Overview of social networks (Koschmider et al., 2010, p. 9)

source	nodes	arcs
Process model	org. units	strength between org. units
User history	users	similarity between users
Insertion history	users	insertion history in terms of users

The strength of the relationship between organizational units enables users to consider the fitness between two process model parts which might be executed in different organizational units or departments and by different groups of people or roles. The social network can thereby be seen by the modeler to verify the existing connections between these and choose a more fitting recommendation. The other social networks register the modelers changes during the creation of a new process model part by editing an existing one or combining some parts. It supports reusing this history for similar users in order to faster complete an editing process and propose changes to an existing process part depending on the insertion history. The similarity between users is calculated by a matrix distance measurement based on the history of selecting certain process model parts. Choosing the same recommended process model parts creates a stronger connection between the users and influences thereby the proposed recommendations in the future modeling processes.

The authors prepare a real-world experiment to proof their success in using such social networks to provide better support for the modeling user. Therefore, they populated a repository with syntactical correct business process model parts and an artificial social network which resulted from a prior questionnaire. The created social network was used to annotate the process model parts individually for each participant. During the modeling process the recommended fragments were attached with the names of people who used them beforehand. These names were from people the participant knows well, knows a little bit or doesn't know at all. The participants were not aware of the fact, that this social connection was artificially inserted and not observed by the system itself.

The result of their experiment is different from what the authors expected to observe. While the modelers are inclined to follow up recommendations from the system they create process models with a higher semantic quality than without, yet, there is no significant value measured in terms of faster modeling, higher syntactical quality or a connection between selecting process parts from people they know better compared to parts from people they don't know well or without any information on previous usage. (Koschmider et al., 2010)

Evaluation

The above summarized recommendation system includes a new way to enrich recommended business process fragments with social features. The authors describe very detailed how they set up their system and how they analyzed social networks within the recommendation module. Also, the authors made a huge effort to investigate their hypotheses about the effectiveness of a recommendation system providing such social additions. The setup and the outcome of the experiment itself are described in great detail. Unfortunately, the result of their study was not to their satisfaction. They did not find a significant impact on the recommended parts with social features. The experiment however did miss a control group, which would have been necessary to validate the outcome. While the authors did not find an impact on the social features they proposed, it remains unclear if other social properties might have. Instead of showing the name of people who used a process fragment before, it could be useful to know how experienced the modeler is. This, and other features could be investigated in future research on this topic. The authors also state, that the fact of artificially annotating the proposed model parts in the experiment could have affected the outcome. This should be taken into account for further research on this topic.

S. Bobek, M. Baran, K. Kluza, G. Nalepa

Application of Bayesian Networks to Recommendations in Business Process Modeling

2013

Summary

Bobek et al. 2013 present in their work a method that applies Bayesian Networks (BN) for recommendation purposes in business process models. They describe their BN representation, modeling and training issues as followed.

"*Bayesian Network is an acyclic graph that represents dependencies between random variables and provide graphical representation of the probabilistic model. This representation serves as the basis for compactly encoding a complex probability distribution over a high-dimensional space. The most important advantage of Bayesian Network models is that it is possible to directly exploit the graphical representation of BP diagrams, which can be easily translated into such model. Another advantage is that the output of a recommendation is a set of probabilities, which allows for ranking the suggestion from the most probable to the least probable.*" (Bobek et al., 2013, p. 5)

In the authors approach the business process model is first transformed into a configurable model. The business process configuration is a method that allows the identification of similarities between two or more business process models. This technique helps for comparison of processes, managing them in large repositories, refactorizing them and automatically extract cloned fragments in the repository of models. Each node of such a configurable process is then translated into a node in the Bayesian Network. The dependencies of the process are thereby translated directly into the network model. It is hereby possible to even capture indirect dependencies which may arise from the company's characteristics and require background knowledge. This allows for better recommendation accuracy and preserving the BPMN grammar at the same time.

The authors trained their Bayesian Network with the Expectation Maximization algorithm. Therefore, they provided training data for the learning process consisting of configurable business processes serialized into a machine-readable file.

The trained BN is then able to recommend BPMN elements to the user currently modeling a diagram. This includes recommending single elements or even suggesting a group of elements which are highly probable to be modeled next.

A drawback of the authors presented approach is, that the trained Bayesian Network can only observe elements which were present in the training data. When the modeler enters some elements unseen in the learning phase, the network produces unwanted probabilities and comes to a halt. (Bobek et al., 2013)

Evaluation

The above summarized approach by Bobek et al. from 2013 introduces the reader into a recommendation system which applies the machine learning method Bayesian Network. They describe their representation, modeling and training phase of the network very detailed and clear. The developed system is also tested in a scenario prepared form real-world processes to proof their success. Even though, the authors present that their system is working, it is not described how accurate its outcome is. Also, this approach is not able to work around the problem of possible missing elements in the training data which will result in a system failure if it comes up in the currently modeled diagram. This could be a starting point for future research and solving this problem could further improve the described approach.

Y. Li, B. Cao, L. Xu, J. Yin, S. Deng, Y. Yin, Z. Wu

An Efficient Recommendation Method for Improving Business Process Modeling

2014

Summary

The authors Li et al., propose in their work from 2014 a workflow recommendation technique based on a string-edit-distance (SED) calculation which improves process modeling by providing recommendations to extend or complete a business process model under construction. Their approach addresses the problem of calculating the distance between two processes which could contain complex structures such as AND- / OR- split and joins. The proposed method is implemented as a prototype using pattern discovery and providing workflow recommendations. The prototype is later evaluated regarding efficiency and effectiveness in comparison to other methods.

The implementation contains three core elements. The first module preprocesses the data source and prepares appropriate input for the pattern discovery module which is the second core element. The preprocessing module takes business processes which might be modeled in different graph models and remodels these into uniform models which are then mined by a graph-mining-algorithm to identify business process subgraphs. These subgraphs are passed into the second module of pattern discovery. Here, all duplications of business process subgraphs are eliminated to present only unique results. The subgraph is therefore decomposed into the candidate nodes and its upstream subgraph. Upstream subgraphs with a calculated confidence above a certain threshold are then registered into a pattern table. The last module is the workflow recommendation, where distance calculations are used to provide accurate and efficient recommendations. The selected path in the modeled business process is processed in the same way like the stored models to compare the results. The process includes a generation for minimum depth-first-search (DFS) codes which are then compared with the SED result. The results from comparing the stored fragments with the currently edited model are sorted based on their distance values and their respective candidate nodes are selected for recommendation. The underlying framework of these three modules and their data flows is represented in Figure 9.

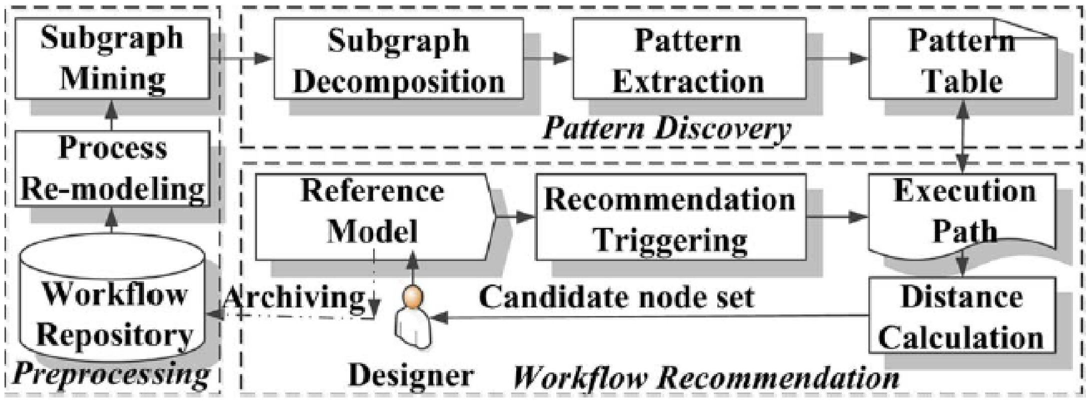

Figure 9 - Framework of graph-based workflow recommendation (Li et al., 2014, p. 507)

The authors compared their SED-based approach with two other approaches including a Graph-Edit-Distance (GED)-based one and FlowRecommender. Therefore, they conducted a comprehensive study using synthetic as well as real-world data sets to study the effectiveness and the efficiency of their implementation. The effectiveness study observes the accuracy the recommender system reaches with a predefined data set. The study reveals, that the accuracy on synthetic sequence datasets, which were needed in order to compare the results with the FlowRecommender system which does not support parallel data sets, is higher for SED-based and GED-based methods. Also, the accuracy is compared in a parallel dataset, where the SED-based method achieves a slightly higher accuracy. This trend can also be observed on a real-world dataset.

For the efficiency study the authors observed the average time for recommending the next node. While FlowRecommender provides the fastest responses on a synthetic sequence dataset, the SED-based method is superior to a GED-based method in this case, and in synthetic parallel and real-world datasets. The reason for FlowRecommender to be the fastest is caused by its limitation of only being able to work with sequential datasets. (Li et al., 2014)

Evaluation

The above summarized work presented by Li et al. 2014 provides great insight to the approach of using depth-first-search codes combined with a string edit distance calculation to provide efficient and accurate recommendations to modelers constructing a business process model. The authors proof that the system is superior to two other approaches in a well-designed and described study. Yet, they do not provide a theoretical evidence that the relationship between SED-based distance calculations and recommendation accuracy is present. This could be an important topic for future research efforts to further validate the usage of this method.

S. Deng, D. Wang, Y. Li, B. Cao, J. Yin, Z. Wu, M. Zhou

A Recommendation System to Facilitate Business Process Modeling

2017

Summary

Deng et al. 2017 identify the following problems in other systems:

They suffer from two main problems: 1) they cannot deal with processes with complex structures, such as AND-split, AND-join, OR-split, OR-join, and cycle, which are common in real business processes and 2) their performance on efficiency and accuracy cannot meet the requirements of practical applications.

In their work the authors propose a process recommendation system which tries to solve these identified problems and help business process analysts to build new processes from scratch in an efficient and accurate way. The proposed approach should be able to help analysts in case of missing domain knowledge or to refine their primary intention and vague idea by providing useful suggestions.

Overall the approach is split into two phases, first is offline mining where the module mines relations among activity nodes from existing processes in a repository, and in the second phase it compares the edited process with the pre-mined patterns to provide proper recommendations of activity nodes with the most matching patterns.

Instead of recommending a complete process model or definition from the repository to the business process analysts in a process discovery and process retrieval technique, the proposed system can recommend proper activity nodes to assist analysts build new process in an interactive way.

In conclusion, compared with the existing work on position-based recommendation, the proposed system is characterized by the authors as the following.

1) Benefited from advanced graph mining techniques, it can extract all relations among process activities efficiently, including explicit and implicit patterns.

2) It provides advanced matching strategies with high efficiency and accuracy to find appropriate node(s).

3) It can enable business process analysts to choose different strategies according to the complexity of processes and balance different requirements in terms of efficiency and accuracy, which improves the applicability.

4) It supports dealing with processes with complex structures such as AND-join/split, XOR-join/split, and cycle, which are very common in real-word business processes.

The proposed system consists of four main components including the user interface, offline mining techniques of process files, a pattern repository which stores the mined patterns and finally the online recommendations as the core of the system.

The user interface enables the import of process definition files for the offline data mining module to extract the patterns as well as to design new processes from scratch with the support of the online recommendation module.

The offline mining component is a key feature of the system. It supports process models which are represented by Petri-net, XML process definition language and business process execution language. It transforms different process models into uniform graph representations and then mines node relations according to three predefined steps: subgraph mining, upstream subgraph decomposition and pattern confidence computation. Finally, the extracted node relations are stored as patterns with the respective properties in a pattern repository.

The pattern repository is used to store and maintain the patterns including the mined properties of upstream subgraphs, candidate node sets and their confidence. The patterns are then provided to the online recommendation module with the function of pattern query, discovery and matching.

Online Recommendation is the core module of the system. It performs a real-time online recommendation based on three different strategies such as maximal common subgraph (MCSub) and minimal common subgraph distance (MCSD), extended graph edit distance (xGED) and extended string edit distance (xSED). Business process analysts can choose between these strategies to select their most fitting preference on efficiency and accuracy. The authors provide a general performance comparison of these three strategies in the following Table 4.

Table 4 - Performance comparison of MCSD, xGED, and xSED (Deng et al., 2017, p. 1392)

Method	**Simple Process Accuracy**	**Complex Process Accuracy**	**Time Complexity**	**Efficiency**
MCSD	**high**	low	$O(n+e)$	**high**
xGED	low	**high**	$O(n^3)$	low
xSED	middle	middle	$O(n^2)$	middle

This module loops through four predefined steps until the building of the business process model is completed. These steps include obtaining the reference model and the current modeling position from the user interface. It determines the reference subpath of the model according to its position and then computes the similarity between the stored patterns and the reference subpath to be able to recommend the top *n* items to the modeling business process analyst.

The authors conducted a series of detailed evaluations of their proposed system including accuracy, hitrates, precision and performance comparing their own mining strategies against each other as well as a comparison of their proposed system against two other approaches which are FlowRecommender presented by Zhang et al., 2009 and NMSF proposed by Cao, Yin, Li, & Deng, 2013. The experiments are based on two different datasets, a real dataset containing 221 processes collected from a district government including complex structures such as AND- / OR- splits and joins. The other dataset is synthetic with a different number of processes ranging from 200 to 1000 and it does not include these complex structures.

The authors state, that their approach outperforms both other systems in most cases caused by better distance calculation strategies and recommendation methods. Additionally, the authors verify, that the MCSD measuring has the best performance on the synthetic dataset where no complex structures are present. Yet, the performance advantage shrinks as the amount of stored processes rise in the pattern repository. (Deng et al., 2017)

Evaluation

The above summarized process recommendation system presented by Deng et al. from 2017 can assist business process analysts by recommending proper fragments based on patterns mined from existing process repositories. The strategy to select fitting process fragments can be chosen by the modeler to properly fit their actual situation and their requirements in terms of efficiency and accuracy. Hereby, the authors provide detailed insight into the results of their conducted experiments on different datasets and chosen recommendation strategies. They display their outcomings in clear and understandable figures and graphs. The applied techniques, formulas and algorithms to identify fitting process fragments and to data-mine existing datasets are explained very explicitly and clear. The authors plan to even further improve the recommendation strategies by combining the measured confidence and the distance together via advanced techniques, incorporating more useful information, such as the function or description of activity nodes, and adopting advanced matching and recommendation techniques. For future research directions it is possible to try to utilize information from process event-logs or additional process model information to acquire more useful patterns for the recommendations.

4.2 Subject-Based Recommendations

The following chapter includes recommendation systems which aim to help the user fulfill a certain degree of fitness, syntactical correctness and structural correctness. Thereby, the suggestion itself might be independent from the current modeling position and rather focusses on suggestions for missing attachments, links between elements, structural recommendations like missing flows or joins and finally textual recommendations to meet organizational standards in naming. Subject-based recommendation techniques require a profound knowledge of contextual information regarding the organizational standards, rules and interconnections to be of any help.

K. Kluza, M. Baran, S. Bobek, G. Nalepa

Overview of Recommendation Techniques in Business Process Modeling

2013

Summary

Kluza, K., Baran, M., Bobek, S., Nalepa 2013 do not present a recommendation system in their work. Yet, they contribute in providing an overview of recommendation possibilities for business process models based on several machine learning methods to recommend business process model fragments. They analyze the possible application of machine learning methods to a recommendation system and provide a comprehensive summary of these methods and their support for the different recommendation fields which are already covered in chapter 2.9 and 2.10.

Their main result is displayed in Table 5.

- The black circle denotes full support of particular machine learning method to recommendation.
- The half black circle denotes partial support of particular machine learning method to recommendation.
- The empty circle denotes no or very limited support of particular machine learning method to recommendation.

Table 5 - Comparison of different machine learning methods for recommending features (Kluza et al., 2013, p. 53)

	Clustering algorithms[a]	**Decision trees**[b]	**Bayesian networks**[c]	**Markov chains**
Attachment recommendations	❍	●	❍	❍
Structural recommendations	❍	●	●	●
Textual recommendations	❍	❍	◗	●
Position based classification	❍	●	●	●

a) Useless as an individual recommendation mechanism, but can boost recommendation when combined with other methods

b) No cycles in diagram

c) No cycles in diagram

The authors find, that the clustering algorithm "*requires developing methods for feature extraction from BPMN diagrams, which is not trivial and still an unsolved task.*" (Kluza, K., Baran, M., Bobek, S., Nalepa, 2013, p.53-54). Nevertheless, they state, that the combination with other methods can be very useful.

The use of decision tress provides a powerful tool for classification that exploits the tree data structure to represent data. Given that a great number of previously build diagrams can be used for learning, it is possible to build a tree that can be used for predicting the next possible element of a BPMN diagram. But this is not possible for diagrams which include cycles due to the nature of the trees structure.

Using the probabilistic graphical model Bayesian networks enables the recommendations to be ranked according to their probability of occurrence. But this method will not be efficient for large diagrams due to the exact inference in Bayesian networks, the authors state. Selecting small chunks of a BPMN diagram could solve this problem.

The characteristics of the Markov chains allow for cycles in the BPMN diagrams processed. The resulting recommendations are based on the probability of the next occurring element. (Kluza et al., 2013)

Evaluation

Despite the fact, that the authors do not present a system for business process recommendations, they provide a detailed insight and analysis into possible machine learning methods for different recommendation features for future developments or research.

A. Koschmider, T. Hornung, A. Oberweis

Recommendation-based editor for business process modeling

2011

Summary

Koschmieder, A. et al. present in their work from 2011 a system for supporting users during the modeling process by providing recommendations from process models stored in a repository through a search interface. The recommendations resulting from the specified query are ranked based on mandatory and optional criteria which are set by the user. For this, they use a modified version of a Term and Document Frequency measure to match the query with stored meta data of process models in the repository. Also, they consider the number of previous reuses of a model fragment and the number of operations performed on the selected recommendation afterwards. The user can optionally add structural correctness, cost and quality of the process design as ranking criteria.

The authors approach includes an automatic tagging mechanism with the purpose of unveiling the modelers intention during the modeling process to better meet the user’s model requirements. Therefore, every business process model stored in the repository is associated with metadata. This metadata is then used for the recommendation and search functionality. While it is straightforward to identify certain keywords in text documents through highest frequency, it is less obvious how to identify them in business process models. The authors came up with an approach, where the description that specifies the purpose of a certain process activity or state is used as a tag candidate for certain process model parts or items. These describing attributes are then later searchable through the query interface. To overcome the limitation of users using different vocabular to describe the same thing, each keyword is also extended with a set of synonyms of equal ranking. Above the automatic tagging mechanism, the modeler can add additional metadata manually including: process name, purpose, objective description, process description, property. Also, the user can annotate the model activity with cost for the design and the quality of the design of a process activity.

The user is later able to search for complete process models, fragments or both using a basic search term or extended search queries. For the extended query interface additional parameters can be set to further limit the query results. This includes also the search for fragments which would not lead to structural errors.

The process recommendations are ranked according to their frequency score, describing how often this process model has been selected before, operation score, indicating the number of edits made after selecting this recommendation, its design score and design quality.

The authors investigate the usefulness and efficiency of their approach with an empirical study. Therefore, they logged all actions performed by the participants during their work with the recommendation system. Additionally, the participants were questioned to justify their decisions.

One main result of their study was, that all participants showed a strong willingness for using the recommendation support system. While the users frequently interacted with the recommendation system, the authors could not demonstrate that the modeling process was faster compared to traditional modeling. This could be based on the fact, that the participants were not well experienced in the usage of this recommendation system. But the authors found, that the selected recommendations are strongly influenced by the semantics of the process element names. (Koschmider et al., 2011)

Evaluation

The above summarized work from Koschmieder et al. from 2011 present a subject-based recommendation system in a very detailed and empirically validated way. The authors provide insight into their measuring processes and used algorithms to score certain properties for a ranking. All the used methods are reasoned and explained in great detail to the reader. Also, the empirical study is described explicitly to validate the results. Yet, one of their key findings might not be applicable for real world use cases, namely the modeling time, which would not decrease by using their proposed recommendation system. The result of this might change, when the users are more comfortable and experience with the use of the system. This effect could potentially be seen in other scenarios as well and is worth a future research effort.

4.3 Other-Based Recommendations

While most of the recommendation systems are based either on position or subject of the modeling context, it might appear that authors propose another base for the selection of fitting recommendation elements. These systems are analyzed in this chapter.

N. Clever, J. Holler, M. Shitkova, J. Becker

Towards Auto-Suggested Process Modeling – Prototypical Development of an Auto-Suggest Component for Process Modeling Tools

2013

Summary

The authors Clever, Holler, Shitkova, & Becker present in their work from 2013 a component which provides model language independent suggestions. The implementation of their work is realized as a web application containing an interface to communicate with. They describe the different architectural approaches one could follow to provide this functionality including a direct integration of the component into an existing process modeling environment or a more generic approach where the component is environment-independent and either based on a specific language or not. The authors chose an architecture for their component which includes a self-contained and language-independent approach which enables it to be used in a much broader context and no limits to language specific characteristics.

For the suggestions itself, the authors identified a set of five criteria which are relevant for the selection of the recommended fragments, including frequency of the sequence, date of insertion, previous users of the suggestion, frequency of suggestion-adoptions and similarity of element labels. The basis of their suggestion component is a knowledge database which stores process models as predecessor-successor-pairs. Despite the fact, that the authors declared the above-mentioned criteria as relevant, they chose a rather basic selection methodology for their prototype implementation. Here, they only take two aspects into their calculation which is frequency of usage and the date of the last insertion. They calculate a score for each possible suggestion based on these properties and present a fixed amount of them in a ranked order to the modeler. Unfortunately, the provided suggestions of their prototype might not fit into the current modeling context, yet they state this downside is more acceptable than presenting an excess supply of suggestions.

In order to connect a modeling environment to their auto suggest component, they provide an interface to exchange the learning input on the one hand and receive the suggestions on the other hand.

The architecture for their component as well as data exchange flows are displayed in Figure 10.

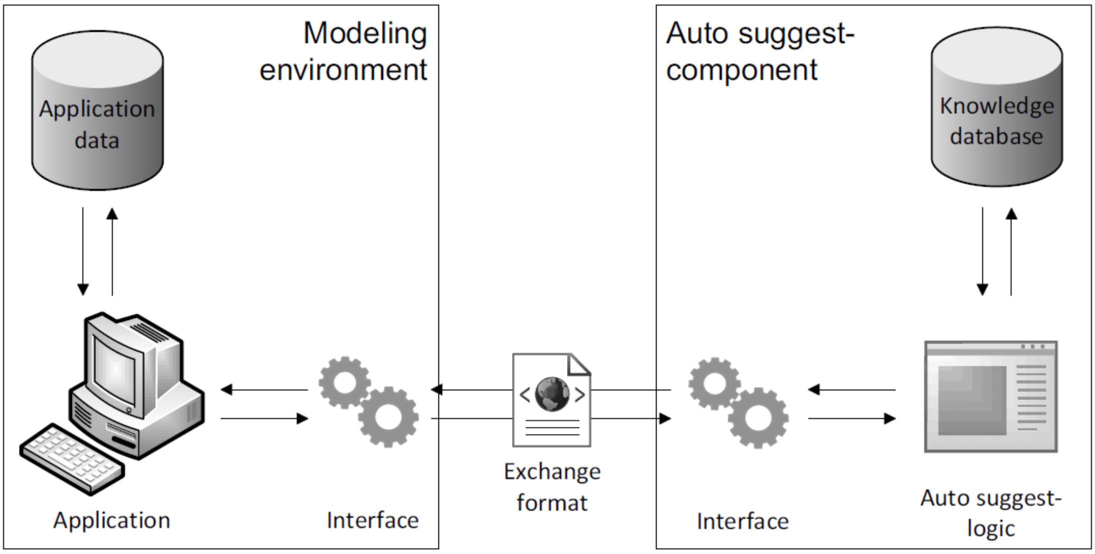

Figure 10 - Architecture of the auto-suggest component (Clever et al., 2013, p. 141)

Three requirements were identified for the data exchange format between the model environment and the auto suggest component. These are:

- Easy to implement for various modeling environments
- Powerful enough to transfer all required information
- Short processing time to securing acceptable performance of suggestions

The chosen data exchange format for their prototype was therefore JSON (JavaScript Object Notation). Based on this data exchange, the applying modeling environment has to provide the following functionality:

- Obtain suggestions: request a suggestion from the component via the interface
- Maintain connection: provide the data set for initial knowledge database setup, provide changes made during the modeling process, provide information about insertion of new connections
- Display suggestions: the modeling environment has to display the received suggestions from the auto complete component
- Apply suggestions: after selection of a suggestion by the user, the modeling environment must apply them

If the modeling environment provides the required functionality mentioned above, the auto suggest component will be able to interact with it to provide recommendations during the modeling process. (Clever et al., 2013)

Evaluation

The above summarized work from Clever et al. from 2013 provides a good insight into the authors research effort to identify key features and requirements for recommendation systems in a process modeling context. They also discuss the advantages and disadvantages of choosing different implementation scenarios for such a component. Finally, they present a prototype implementation of their work choosing one of their discussed scenarios and a set of properties which the suggestion scoring calculation uses to provide recommendations. The decision to implement the prototype in the most independent way proposed, namely, independent from the modeling environment and language independent, might be a good effort for future research and implementations. Yet, the authors only chose a minor set of properties to calculate the score of a fitting recommendation during the modeling process. Only the frequency of usage and the date of last usage are considered here. This misses the point of a context aware recommendation and therefore, possibly provides fragments which are of no use for the modeler. While the implementation effort is a proof of concept for their architecture, the applied recommendation method needs further improvement to provide become a helpful assistant for the user.

5 Concept Matrix

The previously summarized and analyzed literature is finally put into a concept matrix as described in chapter 3.1. The described key features as well as applied recommendation methods are highlighted for each work in Table 6 and Table 7. Also, a combination of both is presented in Table 8. The Literature is ordered descending by the year of release and alphabetically by the authors names.

5.1 Key Features

Table 6 shows the constructed concept matrix including the authors proposed key features. The identified key features are:

- **Store fragment in repository**: In the proposed implementation it is mentioned, that the recommended elements are part of a business process model stored in a process repository.
- **Recommend Fragments**: The proposed recommender system is able to recommend more than a single node. It can recommend a composition of many nodes which are called fragments.
- **Store nodes in pattern table**: The pre-processed models are analyzed regarding a certain pattern. Together with the pattern a candidate node is stored in this table. The pattern is used to compare with the current context and the node is the one recommended to the user.
- **Recommend nodes**: The described system can recommend a single node to the user currently editing the model.
- **Textual recommendations**: The recommendations include a textual component for naming certain elements appropriate to the organizational standards.
- **Aware of current modeling context**: The selection criteria of fitting elements to recommend include an awareness of the current modeling context, already modeled elements or by the user selected elements which are then used for comparison to find a fitting recommendation.

- **Aware of level of abstraction**: In the process for selecting a fitting element for recommendation the level of abstraction of the whole process model is considered.
- **Aware of semantic similarity:** Recommendations are based on the semantic similarity between already named elements and fragments in the process repository.
- **Aware of user experience:** The recommendation process considers the user experience of process models stored in a repository to rank recommendations higher if the modeled elements are made by a user with higher modeling experience.
- **Forward- and backward oriented modeling:** The system is able to provide recommendations for a modeling process starting at a start node as well as the ones modeled backwards where the modeling process starts at the end.
- **Support complex structures:** The recommendations as well as the processing of previous models can include complex structures. Complex structures include AND- / OR- split and join as well as cycles inside the process flow.
- **Social recommendation features:** The recommendations made by the system are enriched by annotations regarding social properties. The recommended elements show properties such as the name of the creator, the name of the last editor, people who used this element lately, date of creation and last usage or organizational units in which this fragment was modeled in.

The **‘x’** in the concept matrix determines, that the proposed work or implementation includes these key features.

Table 6 - Concept Matrix Key Features

Literature		Key Features											
Author	Year	Store fragments in repository	Recommend Fragments	Store nodes in pattern table	Recommend Nodes	Textual recommendations	Aware of current modeling context	Aware of level of abstraction	Aware of semantic similarity	Aware of user experience	Forward- and backward oriented modeling	Support complex structures	Social recommendation features
Deng, S. et al.	2017	x	x		x		x					x	
Li, Y. et al.	2014		x	x	x		x					x	
Bobek, S. et al.	2013		x		x		x					x	
Clever, N. et al.	2013	x	x									x	
Kluza, K. et al.	2013						x						
Koschmieder, A. et al.	2011	x	x				x		x			x	x
Koschmieder, A. et al.	2010	x	x		x		x						x
Hornung, T. et al.	2009	x	x					x	x	x	x	x	
Zhang, J. et al.	2009			x	x		x						

5.2 Recommendation Methods

In Table 7 the different methods used by the authors to determine the most fitting element for recommendation is classified. The recommendation methods range from different graph edit distance measuring algorithms towards neural networks trained by existing data sets. Also, the recommendation methods are classified by their context base similar to the classification used in chapter 4 into position-based, subject-based and other-based recommendations.

The **'x'** in the concept matrix determines, that the proposed work or implementation uses this recommendation method to realize the marked key features.

Table 7 - Concept Matrix Recommendation Methods

Literature		Recommendation Methods															
Author	Year	maximal common subgraph distance	minimal common subgraph distance	(extended) graph edit distance	(extended) string edit distance	Minkowski distance	Hamming distance	Pearsons's correlation coefficient	Sub-path probability	Clustering algorithms	Decision tree	Bayesian networks	Markov chains	Refinemenet Pattern	Position-based recommendation	Subject-based recommendation	Usage-based recommendations
Deng, S. et al.	2017	x	x	x	x										x		
Li, Y. et al.	2014			x	x										x		
Bobek, S. et al.	2013											x			x		
Clever, N. et al.	2013																x
Kluza, K. et al.	2013									x	x	x	x		x	x	
Koschmieder, A. et al.	2011															x	
Koschmieder, A. et al.	2010					x	x	x							x		
Hornung, T. et al.	2009													x	x		
Zhang, J. et al.	2009								x						x		

5.3 Combined

Table 8 shows a combination of both concept matrixes for a better overview of the combinations between key features and recommendation methods since they depend on each other.

Table 8 - Concept Matrix combined

	Literature: Author / Year	Deng, S. et al. 2017	Li, Y. et al. 2014	Bobek, S. et al. 2013	Clever, N. et al. 2013	Kluza, K. et al. 2013	Koschmieder, A. et al. 2011	Koschmieder, A. et al. 2010	Hornung, T. et al. 2009	Zhang, J. et al. 2009
Key Features	Store fragments in repository	×			×		×	×	×	
	Recommend Fragments	×	×	×	×		×	×	×	
	Store nodes in pattern table		×							×
	Recommend Nodes	×	×	×				×		×
	Textual recommendations									
	Aware of current modeling context	×	×	×		×	×	×		×
	Aware of level of abstraction								×	
	Aware of semantic similarity						×		×	
	Aware of user experience								×	
	Forward- and backward oriented modeling								×	
	Support complex structures	×	×	×	×		×		×	
	Social recommendation features						×	×		
Recommendation Methods	maximal common subgraph distance	×								
	minimal common subgraph distance	×								
	(extended) graph edit distance	×	×							
	(extended) string edit distance	×	×							
	Minkowski distance							×		
	Hamming distance							×		
	Pearsons's correlation coefficient							×		
	Sub-path probability									×
	Clustering algorithms					×				
	Decision tree					×				
	Bayesian networks			×		×				
	Markov chains					×				
	Refinemenet Pattern								×	
	Position-based recommendation	×	×	×		×		×	×	×
	Subject-based recommendation					×	×			
	Usage-based recommendations				×					

6 Conclusion and Research perspective

6.1 Summary

The implementation of systems providing recommendation features for process modeling is and remains a challenge. However, these very systems are a great advantage for the modelers and their resulting model. In order for developers and researchers to be able to face this challenge with growing success in the future, it is essential to learn from and correct the mistakes made in previous work. It is equally important to acquire new people for this challenge.

This work covers these two important aspects. Among other things, the introduction highlighted the relevance of this discipline and specific problems in order to attract the attention of potential developers and researchers.

In order to further optimize the efficiency and performance as well as the area of application of these systems, more and more ways are integrated and tested to be able to solve even complex problems in the recommendation scenarios.

In order to make it easier for future researchers and developers to get started, a sub-goal of this work is to explain more about related and relevant topics. Therefore, the second chapter of this work was dedicated to informing the reader in a brief way about the most relevant topics in the area of business process modeling, including the relevance of the popular business process modeling language BPMN as well as challenging yet crucial parts of recommendation systems like level of abstraction of a process model or its recommendation basis. The difference in the basis of recommendations is further used in this work to separate the analyzed literature to ease the comparison between the implementation approaches.

The next section describes in detail how the literature research carried out in this work was defined. First of all, the relevance of a literature search through the work of Webster and Watson was proven. Subsequently, an exemplary overview of the following structuring and classification was given. The framework used delineates the research tools and the search phrases used to search these.

The fourth section of this thesis referred directly to the found literature, which deals with recommendation systems in the area of business process modeling. The aim of this section was to achieve the other two objectives of this work, to highlight the current state of research in the field described as well as problems and solutions. With the conditions which are defined for the literature search, nine research contributions could be identified to be analyzed profoundly in this section.

It is noticeable, that the huge majority of analyzed systems for the purpose of recommendations during the process modeling task are position-based systems. Due to the fact, that the implementation of such systems is rather easy compared to subject-based system requirements into organizational insights, this might not be surprising. These systems classified as position-based share the fact of storing their process models available for suggestions in some kind of repository. This repository could be a basic data storage or a pattern table containing pre-processed models. It is worth mentioning here, that Li, Y. et al. 2014 provided insight into the fact, that pre-processing the models and storing them in a pattern table leads to no significant rise in response time for growing data sets which results in an efficiently working system even if the model repository grows during time. The results of the analysis show, that the latest proposed systems are more likely to support complex structures. The possibility to work with complex structures like AND- / OR- splits and joins in data sets as well as recommendations became more relevant lately due to the limitations for the usage in real world applications. While mostly any of the analyzed systems and proposals are able to maintain awareness of the current modeling context, just one is able to measure the level of abstraction, or user experience and two are able to measure semantic similarity into their selection of fitting recommendations. These properties of a system should not be neglected in future research in this topic since they provide great value for the modeling suggestions. Even though the system may provide fitting elements to the user, they could be even more accurate when the system is able to compare the level of abstraction in the currently modeled process and the ones stored in the process repository. Semantic similarity could lead a way to suggest fragments which fit the modeled process more accurately due to the better understanding of the actual described context. More experienced users are likely to create more accurate and precise models. This awareness could provide recommendations which share their knowledge better with the assisted user. These three described key features are currently not acknowledged by the latest here analyzed research but should make a reappearance in the future to improve the recommendation systems in accuracy, precision and correctness.

The described recommendation methods differ in many aspects. It is noticeable, that the latest research depends on string and graph edit distance rather than probability measures. Also, a neural network made a promising appearance. While both methods appear to be working very efficiently and accurate, a direct comparison between those is not provided in the analyzed literature. It is worth mentioning here, that the usage of a neural network is based on the accuracy and correctness of its training data sets. Its output will not overtake the provided training data and is therefore highly dependent on it. The string- and graph-edit distance measures on the other hand will not be affected by this and could therefore be superior. Yet, this needs to be validated in further research.

The consolidation of the analyzed key features and recommendation methods results in a very diverse set of combinations. The current literature seems to be undefined about the best approaches to provide key features using a certain recommendation method. Further research providing in depth comparison between certain methods or the benefit of implementing a certain feature is needed.

6.2 Limitations

A major problem in creating this work was identifying relevant research contributions. As already described in the methodology, the authors do not use a consistent term to make it clear that the presented work covers the topic of actual recommendation system to assist modelers during a process model creation. The description of an implementation approach or applied techniques to fulfill identified requirements is often neglected in the literature which makes it difficult to understand whether the described system is relevant for this work. The same applies to the terms of the system or the ability to provide recommendations. The term “recommend” or “process model” is also not always indicated or synonyms are used.

It is therefore not possible to search the specified databases specifically for literature about these systems. As a consequence, it is quite possible that research contributions in these databases describe a recommendation system for business process modeling but were declared as not relevant by the definition of the framework for this research or are not listed in the search results.

6.3 Further Research

To conclude this work, suggestions and opportunities for future research are explicitly mentioned in this chapter. In order to facilitate research in this area and to help future researchers and developers to better and more quickly identify relevant literature contributions, a general keyword for the appliance of recommendation techniques should be encouraged. It is conceivable that the authors already make this clear in the title, abstract or in the keywords of their work. Further, a connection between recommendations for business process modeling and the actual use of a system to do so should be highlighted.

As mentioned already, the pattern table as a concept for pre-processing models for recommendations seems to be superior over other repositories. This needs to be validated and is hereby suggested for future research. Providing a definition for the strength, weaknesses and eventual dependencies of such approach could help for future implementations of such systems.

The neural network and string-/graph- edit distance measures as recommendation methods could be taken into in-depth comparison. It is conceivable, that one over the other provides certain benefits or drawbacks. This could include the direct dependency of the accuracy in provided training data for the neural network. A comparison could point out certain scenarios in which one is superior over the other.

Following this, the recommendation methods could be compared even further on their scale of accuracy, efficiency and effectiveness.

Korschmieder, A. et al. 2010 found, that the effect of providing certain social properties with their recommendations is not of any benefit, it is not clearly defined, which key features are crucial or bring the most benefit. A study regarding the modelers imaginings or suggestions could help to concentrate future implementations towards the most beneficial recommendations.

7 Appendix

7.1 Zusammenfassung in deutscher Sprache

Die Implementierung von Systemen mit Empfehlungen für die Prozessmodellierung ist und bleibt eine Herausforderung. Diese Systeme sind jedoch ein großer Vorteil für die Modellierer und das daraus resultierende Modell. Damit Entwickler und Forscher dieser Herausforderung auch in Zukunft mit wachsendem Erfolg begegnen können, ist es unerlässlich, aus den Fehlern früherer Forschung zu lernen und diese zu korrigieren. Ebenso wichtig ist es, neue Leute für diese Herausforderung zu gewinnen.

Diese Arbeit behandelt diese beiden wichtigen Aspekte. In der Einleitung wurde unter anderem die Relevanz dieser Disziplin und der spezifischen Probleme hervorgehoben, um die Aufmerksamkeit potenzieller Entwickler und Forscher auf sich zu ziehen.

Um die Effizienz und Leistung sowie den Einsatzbereich dieser Systeme weiter zu optimieren, werden immer mehr Wege integriert und erprobt, um auch komplexe Probleme in den Empfehlungsszenarien lösen zu können.

Um zukünftigen Forschern und Entwicklern den Einstieg zu erleichtern, besteht ein Teilziel dieser Arbeit darin, mehr über verwandte und relevante Themen zu erklären. Daher war das zweite Kapitel dieser Arbeit der kurzen Information des Lesers über die wichtigsten Themen im Bereich der Geschäftsprozessmodellierung gewidmet, einschließlich der Relevanz der populären Geschäftsprozessmodellierungssprache BPMN sowie schwieriger aber entscheidender Teile von Empfehlungssysteme wie Abstraktionsgrad eines Prozessmodells oder dessen Empfehlungsgrundlage. Der Unterschied in der Basis der Empfehlungen wird in dieser Arbeit weiter genutzt, um die analysierte Literatur zu trennen und den Vergleich zwischen den Implementierungsansätzen zu erleichtern.

Im nächsten Abschnitt wird detailliert beschrieben, wie die in dieser Arbeit durchgeführte Literaturrecherche definiert wurde. Zunächst wurde die Relevanz einer Literaturrecherche durch die Arbeit von Webster und Watson nachgewiesen. Anschließend wurde ein beispielhafter Überblick über die folgende Strukturierung und Klassifizierung gegeben. Das verwendete Framework beschreibt die Recherchetools und die Suchphrasen, mit denen diese durchsucht wurden.

Der vierte Teil dieser Arbeit bezog sich direkt auf die gefundene Literatur, die sich mit Empfehlungssystemen im Bereich der Geschäftsprozessmodellierung befasst. Ziel dieses Abschnitts war es, die beiden anderen Ziele dieser Arbeit zu erreichen, den aktuellen Forschungsstand auf dem beschriebenen Gebiet sowie Probleme und Lösungen herauszustellen. Mit den Bedingungen, die für die Literaturrecherche definiert wurden, konnten neun Forschungsbeiträge identifiziert werden, die in diesem Abschnitt tiefgehend analysiert wurden.

8 List of Literature

Allweyer, T. (2015). *BPMN 2.0 Business Process Model and Notation - Einführung in den Standard für die Geschäftsprozessmodellierung* (3rd ed.). BOD - Books on Demand, Norderstedt. Retrieved from https://books.google.de/books?id=zqWlCgAAQBAJ&lpg=PA35&ots=705N5p6cHJ&dq=Allweyer%2C T.%3A BPMN 2.0–Business Process Model and Notation%3A Einführung in den Standard für die Geschäftsprozessmodellierung&lr&hl=de&pg=PP1#v=onepage&q=Allweyer, T.: BPMN 2.0–B

Bobek, S., Baran, M., Kluza, K., & Nalepa, G. (2013). Application of Bayesian Networks to Recommendations in Business Process Modeling⋆. *Ceur-Ws.Org*. Retrieved from http://ceur-ws.org/Vol-1101/paper5.pdf

Börger, E. (2012). Approaches to modeling business processes: A critical analysis of BPMN, workflow patterns and YAWL. *Software and Systems Modeling*, *11*(3), 305–318. https://doi.org/10.1007/s10270-011-0214-z

bpmn.org. (2019). Retrieved October 1, 2019, from bpmn.org

Cao, B., Yin, J., Li, Y., & Deng, S. (2013). A maximal common subgraph based method for process retrieval. *Proceedings - IEEE 20th International Conference on Web Services, ICWS 2013*, 316–323. https://doi.org/10.1109/ICWS.2013.50

Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards and Interfaces*, *34*(1), 124–134. https://doi.org/10.1016/j.csi.2011.06.002

Clever, N., Holler, J., Shitkova, M., & Becker, J. (2013). Towards auto-suggested process modeling - Prototypical development of an auto-suggest component for process modeling tools. *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft Fur Informatik (GI)*, *P222*, 133–145.

Deng, S., Wang, D., Li, Y., Cao, B., Yin, J., Wu, Z., & Zhou, M. (2017). A Recommendation System to Facilitate Business Process Modeling. *IEEE Transactions on Cybernetics*, *47*(6), 1380–1394. https://doi.org/10.1109/TCYB.2016.2545688

Ehrig, M., Koschmider, A., & Oberweis, A. (2007). Measuring similarity between semantic business process models. *Conferences in Research and Practice in Information Technology Series*, *67*(May 2014).

Hornung, T., Koschmider, A., & Oberweis, A. (2009). A recommender system for business process models. *Proceedings of the 17th Annual Workshop on Information Technologies & Systems (WITS 2009)*, 79.

International Organization for Standardization/International Electrotechnical Commission, I. O., & others. (2013). ISO/IEC 19510: 2013. *Information Technology--Object Management Group Business Process Model and Notation*.

Kluza, K., Baran, M., Bobek, S., & Nalepa, G. (2013). Overview of recommendation techniques in business process modeling. *Proc. KESE. AGH University of Science and Technology*, 46–57.

Koschmider, A., & Blanchard, E. (2007). User Assistance for Business Process Model Decomposition. *1st International Conference on Research Challenges in Information Science*, 445–454.

Koschmider, A., Hornung, T., & Oberweis, A. (2011). Recommendation-based editor for business process modeling. *Data and Knowledge Engineering*, *70*(6), 483–503. https://doi.org/10.1016/j.datak.2011.02.002

Koschmider, A., Song, M., & Reijers, H. A. (2010). Social software for business process modeling. *Journal of Information Technology*, *25*(3), 308–322. https://doi.org/10.1057/jit.2009.21

Li, Y., Cao, B., Xu, L., Yin, J., Deng, S., Yin, Y., & Wu, Z. (2014). An efficient recommendation method for improving business process modeling. *IEEE Transactions on Industrial Informatics*, *10*(1), 502–513. https://doi.org/10.1109/TII.2013.2258677

Lübbe, A., Allweyer, T., & Schnägelberger, S. (2015). BPM Toolmarktmonitor 2015: Marktübersicht zu BPM-Systeme für Prozessautomatisierung, 31.

Oasis. (2019). Retrieved October 1, 2019, from https://www.oasis-open.org/

Object Management Group. (2014). BPMN Version 2.0.2, (December). Retrieved from http://www.omg.org/spec/BPMN/2.0.2/

Recker, J. (2010). Opportunities and constraints: the current struggle with BPMN. *Business Process Management Journal*, *16*(1), 181–201.

Recker, J., Muehlen, M. Zur, Siau, K., Erickson, J., & Indulska, M. (2009). Measuring method complexity: UML versus BPMN. *15th Americas Conference on Information Systems 2009, AMCIS 2009*, *7*, 4444–4452.

Vom Brocke, J., Simons, A., Niehaves, B., Niehaves, B., Reimer, K., Plattfaut, R., & Cleven, A. (2009). Association for Information Systems AIS Electronic Library (AISeL) RECONSTRUCTING THE GIANT: ON THE IMPORTANCE OF RIGOUR IN DOCUMENTING THE LITERATURE SEARCH PROCESS, 2206–2217. https://doi.org/10.1108/09600031211269721

Webster, J., & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a literature Review. *MIS Quarterly*, *26*(2).

Workflow Management Coalition. (2019). Retrieved October 1, 2019, from http://www.wfmc.org/

Zhang, J., Liu, Q., & Xu, K. (2009). FlowRecommender: A workflow recommendation technique for process provenance. *Conferences in Research and Practice in Information Technology Series*, *101*, 55–61.

zur Muehlen, M. (2008). Getting started with business process modeling. In *IIR BPM Conference, Orlando, Florida*.