Selecting Appropriate Process Models for IT Projects: Towards a Tool-Supported Decision Approach

Michael Dominic Harr
*University of Duisburg-Essen, Germany*, michael.harr@icb.uni-due.de

Sarah Seufert
*University of Duisburg-Essen, Germany*, sarah.seufert@icb.uni-due.de

Follow this and additional works at: [https://aisel.aisnet.org/wi2023](https://aisel.aisnet.org/wi2023)

**Recommended Citation**
[https://aisel.aisnet.org/wi2023/101](https://aisel.aisnet.org/wi2023/101)

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISeL). It has been accepted for inclusion in Wirtschaftsinformatik 2023 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
Selecting Appropriate Process Models for IT Projects: Towards a Tool-Supported Decision Approach

Michael Harr\textsuperscript{1} and Sarah Seufert\textsuperscript{1}

\textsuperscript{1} University of Duisburg-Essen, Faculty of Business Administration and Economics, Essen, Germany
{michael.harr, sarah.seufert}@icb.uni-due.de

Abstract. The appropriate selection of suitable process models plays an important role for IT project success. To aid in decision-making, IT project management literature offers a plethora of decision models for selecting suitable process models, however, hybrid process models are often neglected and adoption in practice is low or non-existent. To address this challenge, we draw on contingency theory to develop and implement a tool-supported decision model for the selection and evaluation of appropriate process models for IT projects, thereby leveraging artificial intelligence and machine learning in the context of a self-enforcing network. Our model provides an objective tool to assess process model suitability. Results from a conducted online survey with project management experts indicate high validity. Therefore, we contribute to the field of IT project management by expanding AI-based decision models for selecting and evaluating process models through extending the range of covered models and implementing inherent weighting of criteria.

Keywords: IT Project Management, Process Models, Decision Model, Self-Enforcing Network, Contingency Theory.

1 Introduction

In recent years, technological advances induced by ongoing digitization in a wide range of disciplines have also had an impact on information technology (IT) project management (Simion et al., 2018), leading to IT projects becoming increasingly complex in terms of their goals and scope (Alami, 2016; Ribeiro et al., 2021). To cope with increasing levels of complexity, uncertainty, and volatile environments (Bianchi et al., 2020), a “logical construct (or architecture)” (Zachman, 1987, p. 276), principles, methods, and an abstract procedure are required (Balzert, 2009). Process models play a crucial role by providing tools and methods to structure the project teams’ tasks (Broy & Kuhrmann, 2021), thereby acting as a systematic blueprint for planning, realization, and monitoring of projects within a standardized framework (Fischer et al., 1998; Wieczorrek & Mertens, 2011). Although there are arguments against the use of process models (e.g., neglecting the human factor in projects (Hardgrave et al., 2003)), literature “has traditionally viewed them as axiomatically appropriate to improving both, the process and product of systems development” (Fitzgerald, 1998, p. 317). This is also
reflected in the widespread use of a wide variety of process models in practice (Kuhrmann & Linssen, 2014), whereby a paradigm shift towards agile approaches is gaining momentum in organizations to react more adequately to volatile and uncertain environments (e.g., Lee & Xia, 2010; Cram & Newell, 2016; Recker et al., 2017). Despite the frequent utilization of process models in practice, most IT projects fail in terms of adherence to timeframe, quality, or requirements (Fenech & De Raffaele, 2013). According to a study by PricewaterhouseCoopers (2014), the average IT project overrun the budget by 27 percent with one out of six projects exceeding the budget by over 200 percent and the timeframe by more than 70 percent (Flyvbjerg & Budzier, 2011). Noticeably high failure rates and studies, indicating that unsuitable process models are a major reason for failure imply, at least in part, deficient decisions on the selection and evaluation of process models in practice (MacCormack & Verganti, 2003; Bianchi et al., 2020). Although no consensus exists on the effects of the appropriate process model selection on IT project success, some studies indicate that the use of suitable process models favors project success (e.g., The Standish Group, 2010; Jorge-Martínez et al., 2022) while the use of unsuitable practices leads to poorer project management and overall satisfaction with project outcomes (Cooper, 2007; Shenhar & Dvir, 2007). Due to IT projects’ unique environments and dynamic contexts leading to the absence of one-fits-all solutions (Howell et al., 2010), a plethora of decision models to assist in the selection of appropriate process models were developed in literature. However, adoption in organizations is slow or almost non-existent (Fitzgerald, 1998; Albers, 2021). Hence, Shapiro et al. (2007, p. 249) ascribe a recognized discrepancy referred to as “lost in translation” gap. Reasons can be identified both within the organizations themselves as well as in the shortcomings of existing decision models. On the one hand, due to resource constraints, selection is based on the advice of consultants seeking to sell their own approaches (Vavpotič et al., 2004). Further reasons are compliance with certificates (Klüver & Klüver, 2015) or retention of the status quo (e.g., Samuelson & Zeckhauser, 1988). On the other hand, existing decision models constitute, apart from a few exceptions (e.g., Klüver & Klüver, 2015; Albers, 2021), theoretical approaches without practical implementation, rendering their utilization within project organizations challenging.

Drawing on these observations, our research question is “how can the decision-making capabilities of organizations be enhanced for the selection and evaluation of process models for IT projects?”

To answer the research question, we develop and implement a tool-assisted decision model that leverages artificial intelligence (AI) and machine learning (ML) and extends existing AI-based decision models for the selection and evaluation of process models for IT projects by considering a wider range of covered process models and introducing a weighting of criteria.

The remainder of this paper is structured as follows: First, we provide a brief introduction to contingency theory, IT project management approaches, and related work regarding existing decision models. Then, we proceed with the description of the methodological approach followed by the development, implementation, and validation of our proposed AI-based decision model. Finally, we reflect on limitations and propose avenues for future research.
2 Conceptual Background and Related Work

2.1 Contingency Theory and Types of IT Projects

Contingency theory proposes that an organization’s survivability and effectiveness depend on the fit to their context (Drazin & de Ven, 1985; Howell et al., 2010). Hence, organizations are contingent on specific factors, which impact organizational characteristics under consideration (Wood, 1979; Howell et al., 2010). Applied to IT project management, contingency theory criticizes “universalism” (Wood, 1979, p. 335) and proposes that IT project management practices and projects “should be tailored to suit its context” (Howell et al., 2010, p. 257). Although project management literature predominantly “assume[d] that all projects are fundamentally similar” (Shenhar, 2001, p. 394), recent research highlights the absence of one-fits-all solutions for IT projects; instead IT projects are dependent on their type and contingent characteristics for the selection of suitable process models and the successful execution of the IT project (Cockburn, 2000; Cicmil & Hodgson, 2006; Howell et al., 2010; Kabir & Rusu, 2016; Lehmann, 2016). Nevertheless, differences between project types are “commonly ignored” (Lehmann, 2016, p. 2) in scientific publications. Taking into account the contingency theory, IT projects can be characterized depending on surrounding contingent factors inherent in contingency dimensions.

Howell et al. (2010) identified five contingency dimensions: uncertainty, complexity, criticality, urgency, and team empowerment. Uncertainty encompasses factors, both known and unknown, whose influence on the IT project is difficult to assess in advance (Alami, 2016; Andersen, 2016). The complexity of an IT project is regarded as the “degree of differentiation and interdependence of project elements” (Howell et al., 2010, p. 258). Criticality refers to the effects and consequences of unforeseeable events (Lindvall et al., 2002; Howell et al., 2010). Urgency refers to the magnitude to which time constraints (e.g., pace and time pressure) matter in project activities (Pearson, 1990; Howell et al., 2010; Geraldi et al., 2011). The last contingency dimension deals with team capabilities in a broader sense, focusing on (internal) team-related factors (e.g., Cockburn, 2000; Lindvall et al., 2002; Boehm & Turner, 2003), as well as (external) organizational factors like the geographic distribution or the organizational culture (e.g., Boehm & Turner, 2003; van Donk & Molloy, 2008).

Drawing on contingent factors above and on existing definitions of IT projects in literature, an IT project can be defined as a non-routine and temporary endeavor that is contingent on environmental factors and deals with the creation of IT artifacts, which is usually undertaken in phases to achieve one or more defined objectives (Wallace, 2015; BSI, 2019; Hughes et al., 2019; Jayakody & Wijayanayake, 2021).

2.2 Approaches to IT Project Management

IT project management is a complex endeavor in which the project manager is required to orchestrate several project aspects (e.g., project processes, stakeholders, or team members) simultaneously to achieve the project’s objectives (Bakker, 2010; PMI, 2021;
Bourdeau & Shuraida, 2022). To date, IT project management is an evolving phenomenon that suffers from plurality and fragmentation (Atkinson, 1999; Kolltveit et al., 2007; Turner et al., 2013). Hence, drawing on Alexander & Davis’ (1991) hierarchy of process models, figure 1 delimitates ambiguous terms in IT project management.

![Delimitation of Ambiguous Terms in IT Project Management](image)

**Figure 1. Delimitation of Ambiguous Terms in IT Project Management**

Following the line of reasoning by Alexander & Davis (1991), the IT project management approach encompasses guiding principles and beliefs and presents an overarching and abstract outline of how a specific IT project is managed, designed, and governed (Iivari et al., 2000; Gemino et al., 2021). Prevailing literature distinguishes between traditional, agile, and hybrid IT project management approaches (Albers, 2021; Azenha et al., 2021; Ciric et al., 2022; Lagstedt et al., 2022). The traditional approach to IT project management reflects a “transformation view of production” (Anantatmula, 2021, p. 16), meaning that inputs into an IT project are transformed by strictly adhering to an initial project plan to fulfill the objectives (Špundak, 2014; Ciric et al., 2022). The agile approach is based on the values of the agile manifesto (e.g., Beck et al., 2001; Fowler & Highsmith, 2001) and differs from the traditional approach by providing more flexibility to change (Anantatmula, 2021; Gemino et al., 2021). The hybrid approach to IT project management is an emerging phenomenon that either combines traditional and agile approaches or traditional and agile practices into an overarching process model (Gemino et al., 2021; Lagstedt et al., 2022). Hybrid approaches thus tend to reflect a construction-based approach that can be tailored to individual needs by combining traditional and agile practices and methods (Jacobson et al., 2007; Tell et al., 2019).

Process models (e.g., Scrum, Waterfall, Kanban) represent a specific instance of an IT project management approach and are defined as organized sets of concepts, methods (or practices), guiding principles, beliefs, and multi-step procedures that can be tailored to specific requirements (Iivari et al., 2000; Gemino et al., 2021; Bakhirkin & Lukin, 2022). They reduce the complexity of IT projects and thus increase the likelihood of IT project success (Terlizzi et al., 2016; Jayakody & Wijayanayake, 2021; Bourdeau & Shuraida, 2022).

IT project management practices constitute precise techniques or procedures to manage and execute a single phase of an IT project or a single aspect of a process model within an IT project (Gemino et al., 2021).
2.3 Decision Models for Selecting and Evaluating Process Models

The prevailing literature shows that the selection and evaluation of appropriate process models are complex and non-trivial tasks, inter alia due to a large number of available process models (e.g., Vavpotič et al., 2004; Kuhrmann & Linsen, 2014), their possibilities to be combined or customized (e.g., Sommerville, 2018; Albers, 2021), and their suitability for organizations with different types of IT projects exhibiting various characteristics (Vavpotič et al., 2004; Klüver & Klüver, 2015). In IT project management research, a range of decision models have already been established to facilitate the selection of suitable process models for IT projects (e.g., Zachman, 1987; Alexander & Davis, 1991; Nouck & Schienmann, 1999; Gräßle et al., 2010; Habermann, 2013; Bakhtouchi & Rahmouni, 2018; Albers, 2021). However, a large proportion of the decision models simplify the selection and evaluation by differentiating between IT project management approaches (level 3), i.e., whether a traditional or an agile approach should be selected (e.g., Little, 2005; Howell et al., 2010; Ahimbisibwe et al., 2017; Butler et al., 2020), or whether a hybrid approach should be constructed that combines traditional and agile practices and methods (e.g., Jacobson et al., 2007; Tell et al., 2019; Papadakis & Tsironis, 2020). Several decision models have been established in the literature to discern and evaluate specific process models for IT projects on the second hierarchy level (e.g., Kettunen & Laanti, 2005; Jain & Chandrasekaran, 2009; Hicdurmaz, 2012; Moyo et al., 2013; Dawson & Dawson, 2014). However, existing approaches in literature either exclusively focus on a single domain (e.g., Vavpotič et al., 2004; Langer et al., 2010), take a specific perspective (e.g., Zachman, 1987), only contrast a few selected process models (e.g., Fitsilis, 2008; Gräßle et al., 2010), or they neglect to take into account emerging hybrid process models to a sufficient extent (Albers, 2021). Therefore, literature lacks comprehensive decision models that encompass all approaches within their decision framework, since only a limited number of identified decision models meet this criterion, as outlined in table 1.

<table>
<thead>
<tr>
<th>Hierarchy</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3: IT Project Management Approach</td>
<td>(Wysocki, 2019; Azenha et al., 2021; Ciric et al., 2022; Lagstedt et al., 2022)</td>
</tr>
<tr>
<td>Level 2: Process Models</td>
<td>(Klüver &amp; Klüver, 2015; Albers, 2021)</td>
</tr>
</tbody>
</table>

While recent approaches by Klüver & Klüver (2015) and Albers (2021) address some of the mentioned shortcomings and leverage AI and ML in the context of a self-enforcing network (SEN), the model lacks inherent weighting of criteria and thus implicitly assumes that each characteristic is equally important for the selection of suitable process models.
3 Method

Drawing on principles of proper modeling and prescriptive decision theory (Schütte, 1998; Laux et al., 2018), an enhanced decision model based on Albers’ (2021) proposed model is developed, implemented, and validated ex-post in three stages.

First, the decision field is established by conducting a literature review regarding existing process models and considered criteria in IT project management research. Second, the decision model is then implemented in a SEN\(^1\) developed by Klüver & Klüver (2015) since it has the advantage that, in contrast to established artificial neural networks, the knowledge within the weighted matrix is not generated arbitrarily, but comprehensively based on known cognitive learning rules (Klüver & Klüver, 2021). Third, to validate our enhanced decision model, we conducted an online survey with project management experts from German-speaking countries. In detail, invitations were sent to experts from an IT consulting company and a leading German automotive manufacturer. Eligible participants included project management experts who had successfully completed a recent IT project and held prominent roles within those IT projects (e.g., project manager). The experts were asked to characterize ex post a recent successfully executed IT project based on the questionnaire by Albers (2021). In line with Ciric et al. (2022) the questionnaire included a self-reporting (subjective) assessment of the process model, the IT project and its characteristics, and the success of the IT project, as perceived by the respondents. The assessment of the characterized IT project’s success is based on Shenhar & Dvir’s (2007) proposed multidimensional framework for determining project success. In total, the online questionnaire consisted of 8 sections in which a total of 144 variables were queried. In total, 9 successfully completed IT projects were characterized by the experts and eligible for further validation. Finally, decision model validation consisted of inserting successful and ex-post characterized IT projects from respondents into the SEN as input vectors to then validate, whether the SEN would have suggested the same approach and/or process model that was actually adopted.

4 Decision Model for the Selection of Process Models

4.1 Development of the Decision Model

Based on identified process models in the context of a conducted literature review (see https://bit.ly/3Q79Fp8), the most frequently mentioned process models applicable to IT projects were included as alternatives in the decision field. In total, \(n = 17\) process models are considered as alternatives within the decision field. Thus, a total of 6 (35.29%) traditional, 9 (52.94%) agile, and 2 (11.76%) standardized hybrid process

\(^{1}\) The SEN is an artificial neural network based on unsupervised learning that encompasses three fundamental components: a semantic matrix ordering relations between process models and their criteria, a neural network, and a component for visualization (Klüver & Klüver, 2015).
models are considered alternatives (see https://bit.ly/44TPpvd). In contrast to Albers (2021), standards and overarching frameworks were loaded into the SEN but neglected in the evaluation. In addition, Albers (2021) declares ScrumBan as a hybrid process model, however, in the context of this paper it is by definition an agile process model, since two agile process models are combined and not a traditional and an agile process model.

Each process model is defined by one or more criteria, resulting in a tuple of attributes (i.e., vector) that characterizes the process model (Laux et al., 2018). Due to the fact that Albers' (2021) proposed decision model is the only one covering all identified contingency dimensions and due to the intention to enable a comparison between Albers' (2021) decision model and the decision model developed in the context of this paper, Albers (2021) criteria were adopted. In total, 15 project-specific (e.g., budget size, goal uncertainty), 37 project management-specific (e.g., change perception, contract relationship), 17 project team-specific (e.g., team location, team commitment), and 12 organizational evaluation criteria (e.g., management style, monetary incentives) are considered, which contain both subjective and objective evaluations (for a complete list see https://bit.ly/43sKUH6).

To evaluate the suitability of process models, Albers’ (2021) carried out an empirical survey with experts from the German-speaking IT project management domain. By evaluating the process models through experts from the field, practice-relevant findings were thus identified, particularly with regard to the subjective criteria, which are based on actual experience and impressions from the field (Albers, 2021). Therefore, we adopted Albers’ (2021) survey results for input into the SEN.

4.2 Implementation of the Decision Model

Considered criteria, their coding within the SEN, the values obtained through Albers’ (2021) expert survey, and the linguistic scores into which the obtained values were recoded were automatically loaded into the SEN as comma-separated values and displayed in the Attribute Editor.

The Attribute Editor has the following configuration options: For attributes, \( r_{\text{def}} \) represents the default value, \( r_{\text{min}} \) and \( r_{\text{max}} \) represent the respective minimum/maximum value of the attribute, and \( n_{\text{min}} \) as well as \( n_{\text{max}} \) represent the left/right interval limits to which the attribute is normalized. The variable \( \mu \) is defined as the standard deviation of the attribute and \( \sigma \) as the expected value of the variable. “Transformation” describes the procedure by which the variables are preprocessed, with normalization, strict normalization, and standardization available for selection (Klüver & Klüver, 2021). The cue validity factor (cvf) is a numerical factor that can strengthen or weaken their importance (Albers, 2021; Klüver & Klüver, 2021). The variables \( v_{\text{min}} \) and \( v_{\text{max}} \) describe the minimum/maximum weight value of the semantic matrix for an attribute. To set up the semantic matrix using the raw data (i.e., \( v_{\text{raw}} \)), further following variables are defined: Let \( O \) be the set of objects, i.e., the process models, which have been evaluated through Albers’ (2021) expert survey. Then, \( O = \{ o_1, o_2, ..., o_m \} \), where \( m = |O| \) for \( \forall m \in \mathbb{N}^+ \). Furthermore, let \( A \) be the set of attributes with \( A = \)
\{a_1, a_2, ..., a_k\}, where \(k = \|A\|\) for \(\forall k \in \mathbb{N}^+\). Let \(F : A \rightarrow \{1.0, 1.5, 2.0\}\) be the application that associates to each attribute \(a_k \in A\) a \(cvf\) for the attribute. In addition, let \(V\) bet the set of matrices with \(V = \{v_{raw}, v_{norm}, v_{sm}\}\). Finally, let the raw semantic matrix \(v_{raw}\) be the matrix \((k \times m)\), expressing the level of affiliation of each attribute to each associated object (Klüver & Klüver, 2021).

For normalization, a bipolar scale ranging from -1 to 1 (i.e., interval [-1;1]) is specified for coding the attributes. That is, for all attributes, \(n_{min} = -1\) and \(n_{max} = 1\) is defined. Proper normalization of the semantic matrix \(v_{raw}\) into \(v_{norm}\) is then done automatically by computing equation (1).

\[
v_{norm} = \frac{v_{raw} - r_{min}}{r_{max} - r_{min}} \cdot (n_{max} - n_{min}) - n_{min}
\] (1)

For the artificial neural network to be generated, the normalized semantic matrix must be transformed into the weight matrix \((v_{sm})\) by drawing on a learning rule (Albers, 2021). In the weight matrix, weights are assigned to the associations between the objects and the attributes (Klüver & Klüver, 2015). The weight of an attribute associated with an object \(w_{oa}\) is initially obtained by applying equation (2):

\[
w_{oa} = c \cdot v_{sm},
\] (2)

where \(c\) represents the learning rate. A learning rate of \(c = 0.1\) is selected, as this is sufficient in most cases (Klüver & Klüver, 2021). The neural network can be easily reconstructed and retraced following the logic that a given weight value of a future time \(w(t + 1)\) can be determined from its value at the previous time adding a delta (\(\Delta w\)), as shown in equation (3).

\[
w(t + 1) = w(t) + \Delta w
\] (3)

The delta (\(\Delta w\)) is calculated by an adapted learning rule, which incorporates the \(cvf's\) into the computation of the weights. We incorporate the self-enforcing rule (SER) as the main learning rule, which is presented in equation (4).

\[
\Delta w = c \cdot v_{sm} \cdot cvf_a
\] (4)

For the classification of the process models, attribute neurons related to the objects are activated externally. The activation function is then used to calculate the values of the terminal activations of the object neurons in the artificial neural network (Klüver & Klüver, 2015, 2021). We chose the enforcing activation function (EAF) presented in equation (5) as it was explicitly developed for the SER by Klüver & Klüver (2015, 2021) and thus supports the self-enforcing dynamics of the artificial neural network.

\[
a_j = \sum_{i=1}^{n} \frac{w_{ij} \cdot a_i}{1 + |w_{ij} \cdot a_i|}
\] (5)

In equation (5), \(a_j\) represents the activation value of the receiving neuron \(j\), \(a_i\) represents the activation value of the sending neuron \(i\), and \(w_{ij}\) represents the weight of the connection between \(i\) and \(j\) (Klüver & Klüver, 2021).
Since criteria exist, which contribute more significantly to selecting a suitable process model than others, different \(cvf\)’s were applied to the attributes based on identified literature and the agile manifesto (Fowler & Highsmith, 2001) as shown in table 2.

### Table 2. Adapted Cvf’s of Important Criteria

<table>
<thead>
<tr>
<th>Criteria (c_a)</th>
<th>Attribute name</th>
<th>(cvf_a)</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c_6)</td>
<td>Novelty / Market uncertainty</td>
<td>2.0</td>
<td>Ratbe et al., 1999; Ahimbisibwe et al., 2017; Butler et al., 2020; Ciric et al., 2022</td>
</tr>
<tr>
<td>(c_{13})</td>
<td>(IT project) complexity</td>
<td>2.0</td>
<td>Ahimbisibwe et al.,</td>
</tr>
<tr>
<td>(c_{15})</td>
<td>(IT product) complexity</td>
<td>2.0</td>
<td>2017; Butler et al.,</td>
</tr>
<tr>
<td>(c_{39})</td>
<td>Requirements volatility</td>
<td>2.0</td>
<td>2020; Ciric et al.,</td>
</tr>
<tr>
<td>(c_{38})</td>
<td>Time of requirements elicitation</td>
<td>1.5</td>
<td>Beck et al., 2001; Fowler &amp; Highsmith, 2001</td>
</tr>
<tr>
<td>(c_{44})</td>
<td>Stakeholder integration</td>
<td>1.5</td>
<td>Fowler &amp; Highsmith,</td>
</tr>
<tr>
<td>(c_{59})</td>
<td>Team’s hierarchical task organization</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>(c_{61})</td>
<td>Team communication culture</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>(c_{62})</td>
<td>Reflection on collaboration</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>(c_{64})</td>
<td>Willingsness to learn and change</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>(c_{67})</td>
<td>Trust within the team</td>
<td>1.5</td>
<td></td>
</tr>
</tbody>
</table>

All other attributes were assigned a \(cvf = 1.0\). Listed authors have demonstrated the significance of these attributes for the decision to select a suitable IT project management approach, which is why the \(cvf\) was doubled, as this illustrates the high relevance of the attributes (Klüver & Klüver, 2021). For attributes whose value has not been empirically validated but whose importance for the selection of agile process models is nonetheless relevant, a slightly increased \(cvf\) was set (\(cvf = 1.5\)) based on the agile manifesto (e.g., Beck et al., 2001; Fowler & Highsmith, 2001).

Accordingly, by computing equations (2) to (5) with the provided \(cvf\)’s in table 2, the final and weighted semantic matrix \(v_{sm}\) is created with a learning rate of \(c = 0.1\) and one iteration.2

### 4.3 Retrospective Validation and Discussion

In total, 9 successfully completed IT projects were characterized by the respondents. A total of four (44.44%) traditional process models, four (44.44%) agile process models, and one (11.11%) hybrid process model were adopted. Regarding the IT projects that adopted traditional process models, Structured Systems Analysis and Design Method (SSADM) was used once and the waterfall model was used three times as the process model. In agile IT projects, Scrum was adopted three times and Feature-Driven Development (FDD) once. The IT project adopting a hybrid approach used Software Development Agile (SoDa) for this purpose. On average, all IT projects were perceived as

---

2 A thorough explanation of all constructs and transformations is beyond the scope of this paper. For further explanation, see Klüver & Klüver (2015, 2021) and Albers (2021).
successful by the respondents with traditional process models (82.01%) being marginally less successful than agile (85.73%) and hybrid (90.00%) ones. Regarding potential outliers, all characterized IT projects are relatively representative and comparable (79.11%) on average. Since the IT projects characterized by the respondents are both successful and comparable, the identified process models could be transferred into the SEN as input vectors. Before, however, the values of each of the criteria had to be recoded in a similar fashion as the values in Albers’ (2021) expert survey. Hence, the obtained values were recoded into the linguistic scores for the input into the SEN. The decision model is considered to be valid if it proposes the same process model for an IT project that has adopted it. Validity is determined numerically by the values of the final activation of the neurons (i.e., provided rankings in the SEN), or else by the Euclidean distances. The validation is conducted following Alexander & Davis’ (1991) hierarchy of process models at the 2nd and 3rd levels. The three best and the single best recommendation of the model are examined in each case, both from the ranking list and from the distance list. The validity percentage of the decision model is calculated for each level.

Regarding the suitability of the first proposed process model for each input vector, the decision model is 100% valid for the recommendation of a suitable approach to IT project management, which means that for each input vector, the decision model recommends a process model that belongs to the same overarching approach to IT project management. Considering the validity on the level of process models, the ranking still suggests 75% of the correct identical process models. Regarding the Euclidean distance, 62.5% of the first recommended process models are correct. Looking at the first three proposed process models, a similar picture emerges: regarding the IT project management approach, the decision model proposes a process model of the appropriate approach based on the rankings in 83.33% of the cases. In 91.67% of the cases, the decision model proposes a process model of the appropriate approach based on the Euclidean distance. For a total of 54.17% of each of the first three recommended process models, the correct process model was suggested based on the ranking or Euclidean distance. In summary, the developed decision model is 100% valid in determining the overarching approach to IT project management. Regarding specific process models, it is quite valid, as it first suggests a suitable process model for an input vector with a probability between 62.5% and 75% - depending on the selected metric.

In an equivalent manner, the validity of Albers’ (2021) decision model was determined and compared to the results of our enhanced decision model. Overall, our decision model performs equally well or better in terms of recommending process models that can be assigned to the same IT project management approach as the process model of the input vector. With respect to the Euclidean distance, the developed decision model performs about 11.11% better than Albers’ (2021) decision model. However, when considering the specific process model recommendation, the developed model

---

3 The following values always refer to the validity without the inclusion of the characterized IT project adopting SSADM, since there were no entries in the semantic matrix for this process model.
performs ambivalent: if only the first recommendation of the decision models is considered, the decision model performs between 10.0%-12.5% worse than Albers’ (2021) decision model. If, on the other hand, the first three process model recommendations for each input vector are considered, it can be seen that the developed model performs equally well or up to 4.17% better than Albers’ (2021) decision model.

Based on the Euclidean distances, the map visualization (figure 2) illustrates the similarities between the IT projects. It is apparent that the evaluation and assessment of the IT projects or process models occurred heterogeneously, as shown in their broad distribution. However, the IT projects characterized by the experts (highlighted in color) are predominantly assigned to the correct corresponding cluster: the three IT projects using Scrum are located within the agile cluster and the three IT projects using the waterfall model are located within the traditional cluster. SoDa - representing a hybrid process model - is located between the traditional and agile clusters, which illustrates the combination of agile and traditional practices. FDD - typically agile - can be classified as a hybrid approach based on the characterization, which reinforces the importance of adapting existing process models to constantly changing contextual factors (Henderson-Sellers & Ralyté, 2010; Joslin & Müller, 2014). This is in line with current research on situational method engineering (SME), which emphasizes that standardized or individually constructed process models must be adapted to changing environmental requirements and project characteristics, for IT projects to be accomplished successfully (Henderson-Sellers & Ralyté, 2010; Gottschalk et al., 2021). Our AI-based decision model incorporates SME principles (e.g., Henderson-Sellers & Ralyté, 2010;
Gottschalk et al., 2021) by drawing on contingency theory to acknowledge situational factors (e.g., market uncertainty), organizational context (e.g., management style), and team capabilities (e.g., team diversity). By aligning with SME principles, our decision model can cater to the unique requirements of individual IT projects, therefore providing more nuanced recommendations and promoting better project outcomes. In addition, IT projects can be re-characterized at any time, for example, if an adaptation of the process model is necessary due to changing requirements. Thus, our decision model can be utilized for standardized traditional, agile, and hybrid process models as well as for constructed or customized process models, since it allows a classification and comparison between current and previous IT projects without the need for reference vectors.

5 Conclusion

We developed and implemented an AI-based and tool-supported decision model for the selection and evaluation of appropriate process models for IT projects. Building on preliminary work by Albers (2021), we developed an enhanced and practically evaluated decision model that can be easily implemented by organizations and that leverages AI and ML to support decision-makers in diverse contexts. Hence, we contribute to the ongoing dialogue on SME and contingency theory by proposing a novel decision model addressing the construction and adaption of process models due to changing contextual factors. Based on attributes identified in the literature that significantly influence the selection of process models, individual attribute weights were implemented, which led to higher overall activation of neurons, especially for the agile process models, thus facilitating a better overall decision. Hence, we contribute to the field of IT project management and SME by extending existing AI-based decision models by extending the range of considered process models and introducing a weighting of criteria.

The present paper is subject to several limitations and restrictions. With regard to the online questionnaire, its validity can be questioned, since the inherent constructs of the questionnaire were not validated and included self-reporting subjective answers. Due to this small sample (9 characterized IT projects), no comprehensive validation can be guaranteed and further research is required to assess the validity of the decision model. Since experts characterized successfully implemented IT projects in the online questionnaire, there is a further threat to validity in that successfully implemented IT projects may also have adopted inappropriate process models. However, it is implicitly assumed in the context of this research paper that successful IT projects have also properly adopted successful process models.

Future research could empirically validate our proposed decision model and adjust modifications that lead to better decision-making. In contrast to the retrospective characterization of IT projects, the validation could be done ex-ante, which would involve having the decision model propose a process model for an IT project, whose success would then be evaluated after completion. By analyzing case studies and real-world examples, future research should investigate the interplay between process model selection and its successful implementation, to support decision-makers towards better project outcomes.
References


