Opening the Black Box: How to Design Intelligent Decision Support Systems for Crowdsourcing

Marcel Rhyn[1], Niklas Leicht[1], Ivo Blohm[1], and Jan Marco Leimeister[1,2]

[1] University of St. Gallen, Institute of Information Management, St. Gallen, Switzerland
{marcel.rhyn, niklas.leicht, ivo.blohm}@unisg.ch
[2] University of Kassel, Research Center for Information Systems Design, Kassel, Germany
{leimeister}@ uni-kassel.de

Abstract. In crowdsourcing, reviewing and evaluating textual data is a latent challenge. While text mining and machine learning represent promising technologies to solve this problem, it is still unclear how information systems based on these technologies (i.e., intelligent decision support systems) should be designed. In this study, we address this gap and develop overarching design requirements, design principles, and design features for intelligent decision support systems in crowdsourcing. The study follows a design science research approach with a cross-industry research consortium comprising 8 organizations. Our results are based on 41 semi-structured interviews, 13 expert workshops with 53 participants, statistical analyses with data from 676 crowdsourcing projects, and 2 field tests. For research, we introduce transparency and control as two additional meta-requirements for intelligent decision support systems and capture seven guiding principles for designing such systems. For practitioners, we describe specific design features that show how to instantiate these principles.

Keywords: Crowdsourcing, Decision Support, Design Science Research

1 Introduction

In crowdsourcing, organizations use open calls to engage large networks of people and collect their solutions, ideas, or feedback to solve a predefined task [1]. The approach offers the opportunity to take advantage of vast amounts of user-generated data and has found widespread adoption in different domains, including innovation management for product development [1] or software development for application testing [2]. However, it represents a latent challenge to review and evaluate crowdsourced data [1, 3]. Piezunka and Dahlander [4] studied how 922 organizations leveraged crowdsourced data and found that they often "fail to harness the full potential of crowdsourcing due to inadequate filtering mechanisms" (p. 876). Google, for example, required almost three years and 3'000 employees to analyze 150'000 ideas submitted to its *Project 10^{100}* [1].

In order to cope with large amounts of user-generated contributions in crowdsourcing, research and practice are increasingly using text mining and machine learning. A

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

https://doi.org/10.30844/wi_2020_a4-rhyn

number of studies have already demonstrated the potential of these algorithms to support the evaluation of ideas [5], the prioritization of software defects [6], or the identification of locations in incident reports [3]. However, these studies have mostly focused on domain-specific instantiations that demonstrate the technical capabilities (e.g., performance, features) of the algorithms. They have focused less on design knowledge that guides the deployment and adoption of text mining and machine learning in full-fledged information systems [7]. Hence, while the technical development of the algorithms is advanced, it is still unclear how intelligent decision support systems based on these algorithms should be designed in crowdsourcing [8]. Appropriate IS designs are crucial for the acceptance and adoption of such systems [9]. To better understand how these systems should be designed, scholars have called for studies to "contribute guidelines for design artifacts" that support decision-making in these contexts [8].

In this study, we aim to close this gap and answer the following research question: What design principles should guide the development of intelligent decision support systems (DSS) in crowdsourcing? Design principles are statements that capture abstract design knowledge and prescribe "what and how to build an artifact in order to achieve a predefined design goal" [10]. To develop these design principles, we followed a design science research approach based on Peffers et al. [11]. Our research was conducted over 3.5 years with a consortium comprising 8 organizations [12]. It took part in three design-and-evaluate iterations that included 41 interviews, 13 workshops with 53 participants, statistical analyses with data from 676 projects, and 2 field tests. In these iterations, we (1) defined design requirements with related design principles and design features for intelligent DSS, (2) developed software prototypes for a formative evaluation, and (3) instantiated them in organizations for a summative evaluation.

The contribution of this study is threefold. First, we extend existing studies in the decision support field, which have mostly focused on the traditional efficiency-effectiveness framework [13–15], and introduce transparency and control as additional meta-requirements when designing intelligent systems. Second, for research on crowdsourcing, we define design principles to guide the development of intelligent DSS. We extend existing literature, which has already examined specific instantiations of text mining and machine learning [e.g., 3, 5, 6], and capture the necessary design knowledge for their deployment in intelligent DSS. Third, we describe specific design features that show how the design requirements and principles can be addressed.

The remainder of this paper is structured as follows: First, we present the theoretical background of our study and review literature on crowdsourcing and decision support. Second, we elaborate on our design science research approach. Third, we reveal the results and discuss their implications for theory and practice. Fourth, we conclude our study by acknowledging its limitations and offering an outlook for future research.

https://doi.org/10.30844/wi_2020_a4-rhyn

2 Related Work

2.1 Crowdsourcing

The fundamental principle of crowdsourcing revolves around the use of an open call through which an organization engages a potentially large and diverse network of people to collect their contributions to a predefined task [1]. Compared to traditional sourcing approaches that rely on only few designated agents (e.g., innovation managers, testing experts), crowdsourcing seeks to mobilize an independent group of contributors to perform these tasks [1]. The approach facilitates the collection and aggregation of data and allows organizations to benefit from a wide range of user-generated contributions [16]. In this study, we focus on crowdsourcing settings that deal with textual contributions. Such contributions may include ideas for innovation management [1] or bug reports and usability feedback in software testing [2]. Given the decentralized nature of crowdsourcing, contributions are often collected through IT platforms [17]. On the one hand, these platforms enable organizations to allocate tasks to a crowd and coordinate their activities. On the other hand, the platforms serve as focal points for organizations to aggregate and retrieve contributions. In this way, the platforms represent the interface between the organizations seeking to broadcast a task and a large number of contributors willing to perform the task [17]. Individuals working at this interface (e.g., product owners, test managers) take a boundary-spanning role as decision-makers for the organizations [18]. They are responsible for processing the data and selecting relevant contributions for the organizations (e.g., implementing an idea or fixing a defect) [16]. However, the quantity and complexity of user-generated data in crowdsourcing often make it impossible for decision-makers to process the data by themselves in an efficient and effective manner. Especially for crowdsourcing platforms that are based on large amounts of unstructured, textual contributions, it is imperative to employ mechanisms that support decision-makers in integrating or selecting relevant contributions [1].

2.2 Decision Support

Decision support is the area of IS research that is concerned with supporting and improving decision-making in organizations [19]. Decision-making is generally defined as a process comprising three distinct phases: (1) a *processing* of informational cues, (2) an *assessment* of possible courses of actions, and (3) a *commitment* to action [20]. In crowdsourcing, such decision-making processes may comprise (1) an initial screening of user-generated ideas, (2) an evaluation of their projected costs and market potential, and (3) a final choice of implementation [21]. However, extant research suggests that cognitive limitations and bounded rationalities constrain decision-makers in their assessment of information during such processes [13]. With increasing information load [22], it becomes more difficult for decision-makers to identify relevant information [23] or to recall prior information and set priorities [24]. Studies also show that their search strategies through data sets become limited and less systematic [25]. Personal DSS are designed to expand human information-processing capabilities

https://doi.org/10.30844/wi_2020_a4-rhyn

and improve their decision-making in such settings of high information load [13]. Traditionally, this has been achieved either by automating standardizable information processing tasks or by defining and ordering the activities for decision-making, i.e., structuring the process and providing recommendations [26, 27]. A basic design for this type of DSS includes components for "(1) sophisticated database management capabilities […], (2) powerful modeling functions […], and (3) powerful, yet simple user interface designs that enable interactive queries, reporting, and graphing functions" [14].

While traditional DSS have mostly focused on structured data, DSS research has recently witnessed a "move toward dealing with massive collections of relatively unstructured data" [28]. In this context, text mining and machine learning are gaining in importance for decision support. Text mining denotes the process of extracting useful information from unstructured, textual data through the exploration of meaningful patterns [29]. For this purpose, unstructured text needs to be preprocessed into a format that is compatible for machines. Afterwards, machine learning provides the means to recognize patterns or extract information. Supervised approaches (e.g., classification) offer ways to assign contributions to predefined classes. Unsupervised approaches (e.g., clustering) are capable of finding relationships and structures in large sets of contributions without predefined classes. DSS based on these technologies are often referred to as *intelligent DSS* [19]. Designing such systems is challenging because the system characteristics (e.g., ability to learn, autonomy) and the user interaction with the system (e.g., perceived loss of control) differ from traditional decision support technologies [30]. This "shift necessitates reconsidering guidelines for the design product and design process associated with such artifacts" [8], as we cannot rely on existing design paradigms and principles. For intelligent DSS in particular, studies show that, if such systems are not well designed, decision-makers are likely to reject their recommendations and refrain from relying on them [9]. Thus, related work in the IS and design science field has called for more research to "contribute guidelines for design artifacts" [8]. Especially in crowdsourcing, little is known on how to design and leverage "computational approaches to evaluate the feedback of crowdsourcing" [7].

3 Design Science Research Approach

Design science research (DSR) represents a well-established approach in IS research that is concerned with the creation of artifacts seeking to extend the boundaries of human and organizational capabilities [31]. In this study, we are concerned with defining design principles for intelligent DSS in crowdsourcing. Design principles are one of the most widely used vehicles to "convey design knowledge that contribute beyond instantiations applicable in a limited use context" [10]. Research typically conceptualizes design principles in conjunction with design requirements and design features [32]. *Design requirements* represent meta-requirements [33] which describe the "generic requirements that any artifact instantiated from this design should meet" [32]. *Design principles* can be defined as statements that prescribe how instantiated

https://doi.org/10.30844/wi_2020_a4-rhyn

artifacts should be built in order to meet its requirements [10, 32]. *Design features* represent specific ways to implement design principles in an actual artifact [32]. Thus, design principles represent the link between overarching design requirements and concrete design features. They are important on three accounts [10]. First, they *abstract* away from specific instantiations (e.g., design features) and capture design knowledge about instances of artifacts that belong to the same class [34]. Second, they communicate essential design knowledge and *prescribe* "what and how to build an artifact in order to achieve a predefined design goal [i.e., a design requirement]" [10]. Third, they *contribute* to more comprehensive design theories, e.g., IS designs for intelligent DSS [35].

3.1 Research Process and Context

In order to systematically develop design requirements, design principles, and design features for intelligent DSS in crowdsourcing, we followed the well-established DSR process proposed by Peffers et al. [11]. This approach synthesizes the common phases of design science research discussed in existing literature [e.g., 31, 33]. **Figure 1** below provides an overview of our process. As design science research represents an iterative and incremental approach [31], we conducted three design-and-evaluate iterations. The data collection and analysis in these iterations is explained in more detail in section 3.2.

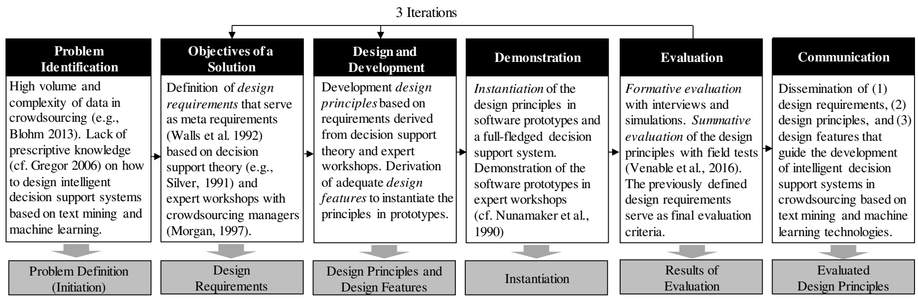

Figure 1. Design Science Research Approach based on Peffers et al. [11]

Our research context was a cross-industry research consortium [12] that consisted of 2 financial institutes, 2 insurance companies, 2 industrial corporations, 1 multinational retailer, and 1 public transportation provider. All companies use crowdsourcing for software testing and innovation (CST). This setting was chosen because CST exhibits two characteristics that make it especially well-suited for developing overarching design principles for intelligent DSS. First, CST comprises different types of textual contributions. Functional testing, for example, requires the crowd to contribute short and technical bug reports with ground truth, while usability testing aims to elicit generative feedback and ideas for software features with no ground truth. Second, CST comprises distinct decision-making tasks. In functional testing, decision-makers are required to judge the severity of bug reports and prioritize them. In usability testing, they need to aggregate feedback and select the most requested features for change

https://doi.org/10.30844/wi_2020_a4-rhyn

requests. Thus, CST can be regarded as a “microcosm” [2] for crowdsourcing insofar that it integrates a variety of textual contributions and decision-making tasks in one unified setting. This should benefit the generalizability of the design principles beyond our research context.

3.2 Data Collection and Analysis

The three DSR iterations were conducted over a period of 3.5 years from 2015 to 2019 and include a total of 41 semi-structured interviews, 13 expert workshops with 53 participants, data from 676 crowdsourcing projects, and 2 field tests with complete DSS.

In the first iteration, we aimed at *defining* an initial set of design requirements, design principles, and design features for intelligent DSS in crowdsourcing. For this purpose, we reviewed existing literature on decision-making and decision support [e.g., 13, 14], and conducted 4 expert workshops and interviews with 40 participants from our research consortium. We asked the participants to describe the crowdsourcing process, explicate focal challenges, and outline potential improvements through DSS. We took notes and clustered the responses. To evaluate our results, we conducted 31 semi-structured interviews with independent, external subject-matter experts (e.g., testing experts, QA managers). These interviews served as a first, formative evaluation[1] [36].

In the second iteration, we focused on *instantiating* the design requirements, design principles, and design features in feasible software prototypes to ensure their technical feasibility. For this purpose, we referred to well-established text mining and machine learning algorithms [e.g., 37]. We developed these prototypes in Python and demonstrated them in 9 expert workshops with a total of 16 participants to gather insights on how to configure the algorithms, achieve sufficient performance (i.e., accuracy, sensitivity, specificity), and visualize the results. To ensure that the prototypes are feasible and achieve the targeted performance, we used training and test data from 676 crowdsourcing projects conducted by organizations in our research consortium. The data comprised more than 300’000 crowdsourced contributions. Training and testing the prototypes served as a second, formative evaluation[1] [36].

In the third iteration, we focused on *implementing* a complete DSS in organizations of our consortium. We used insights from existing research on DSS designs [e.g., 27] to develop a functional frontend, a backend with our text mining and machine learning prototypes, and a database for a web-based DSS in crowdsourcing. Before finalizing and implementing the complete DSS, we conducted 10 semi-structured interviews during which we demonstrated mockups of the system to experts (e.g., testing experts, QA managers) to gather their feedback on the functionality of the system and the visualization of the results. This was done to ensure that the final DSS is suitable and ready to be used by the decision-makers in the organizations. As a concluding, summative evaluation[1], we conducted 2 field tests and implemented the DSS in organizations [36].

[1] For more details on the evaluation, please refer to section 4.2

https://doi.org/10.30844/wi_2020_a4-rhyn

4 Results

4.1 Design Principles for Intelligent DSS in Crowdsourcing

Design principles communicate design knowledge on how to build an artifact to achieve a predefined design goal [10]. We refer to such design goals as design requirements [32, 33]. In decision support theory, existing research typically describes two primary objectives of decision-makers: maximizing decision quality and minimizing effort [13, 14]. In practice, the decision-makers in our workshops and interviews described similar goals and outlined two major issues in crowdsourcing that need to be addressed: (1) the quantity of contributions and (2) the complexity of their content. The former makes the evaluation time-consuming (e.g., "*it is not possible to manually process and evaluate all data*"; Innovation Manager, IT Services). The latter induces a high information load and makes the evaluation error-prone (e.g., "*it is definitely possible that a business analyst will reject [a change request] at a later stage because I made a mistake*"; Test Manager, Retail Bank). Thus, as a first important insight, we find that intelligent DSS should at least aim to increase (1) the efficiency and (2) the effectiveness of decision-making in order to be useful in crowdsourcing. Importantly, however, the workshops and interviews revealed two additional requirements that have received much less attention in existing DSS research: maintaining transparency and control during decision-making. For decision-makers, intelligent DSS often represent a black box if they are not well explained (e.g., "*I would not blindly trust automated reports. I always want to know what is going on. I want to have enough control to be able to intervene"*, Test Manager, Insurance). Transparency can be defined as a "mechanism to expose decision making" [38]. Siau & Wang [39] explain that for intelligent DSS, it is crucial to be able to understand "how they are programmed and what function will be performed in certain conditions. [A DSS] should be able to explain/justify its behaviors and decisions" (p. 51). Control, on the other hand, refers to "the degree of actual influence over the nature of the decision made" [40]. It involves authority over the procedure through which a decision is made. Thus, as a second important insight, we argue that the design of an intelligent DSS in crowdsourcing should be considerate of two additional meta-requirements that revolve around (3) maintaining a sufficient degree of transparency and (4) maintaining a sufficient degree of control by its user (see **Figure 2**).

https://doi.org/10.30844/wi_2020_a4-rhyn

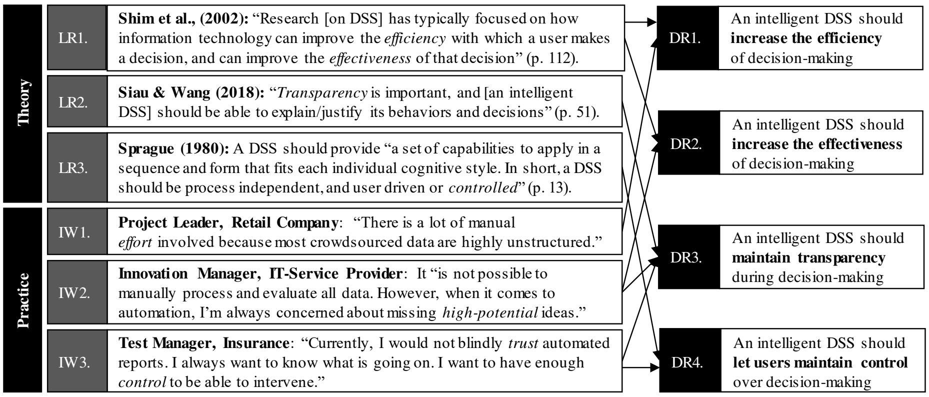

Figure 2. Design Requirements for Intelligent DSS in Crowdsourcing

Guided by these design requirements (DR), we developed design principles (DP) and design features (DF) for intelligent DSS in crowdsourcing. They are informed by moderated expert workshops and interviews in which we asked decision-makers from our consortium to describe their crowdsourcing processes, explicate focal challenges, and outline potential improvements through DSS. We developed design principles and design features, compared and discussed them across the workshops and interviews, and mapped them to the previously defined design requirements. We repeated the process until we reached consensus with regard to the principles and features. In the end, the workshops and interviews revealed 4 design principles that make it possible to reduce the manual effort and information load in crowdsourcing: an *omission* of irrelevant contributions (DP1), a *consolidation* of redundant contributions (DP2), a *prioritization* of important contributions (DP3), and an *indication* of the recommended decision (DP4). DSS that follow these design principles help to increase the efficiency (DR1) and effectiveness (DR2) of decision-making in crowdsourcing. In crowdsourcing, it is possible to instantiate these principles with quality filters (e.g., a classifier that identifies high and low quality contributions based on textual features, such as their length or the number of spelling mistakes; DF1), triaging systems that group similar feedback (e.g., a classifier that is trained to assign contributions to a category; DF2), duplicate detection with sentiment analysis (e.g., a classifier that automatically identifies critical reports; DF3), and recommendations (e.g., a classifier that calculates probabilities for successful implementation based on past data; DF4). However, given that these features are part of intelligent DSS and build upon text mining and machine learning algorithms, it is crucial to maintain transparency (DR3) and control (DR4). The workshops and interviews revealed three additional design principles to address these design requirements: a *translation* of machine outputs in human understandable actions (DP5), an *explanation* of operations leading to recommended actions (DP6), and a potential *adaptation* of operations and rules (DP7). To instantiate these principles [e.g., 38], the DSS should communicate the results in actionable and easy interpretable statements instead of abstract values or outputs (DF5), include popups and tooltips to explain the results (DF6), and allow the user of the

https://doi.org/10.30844/wi_2020_a4-rhyn

DSS to configure the system and control the workflow (DF7). **Figure 3** provides an integrated overview of our findings.

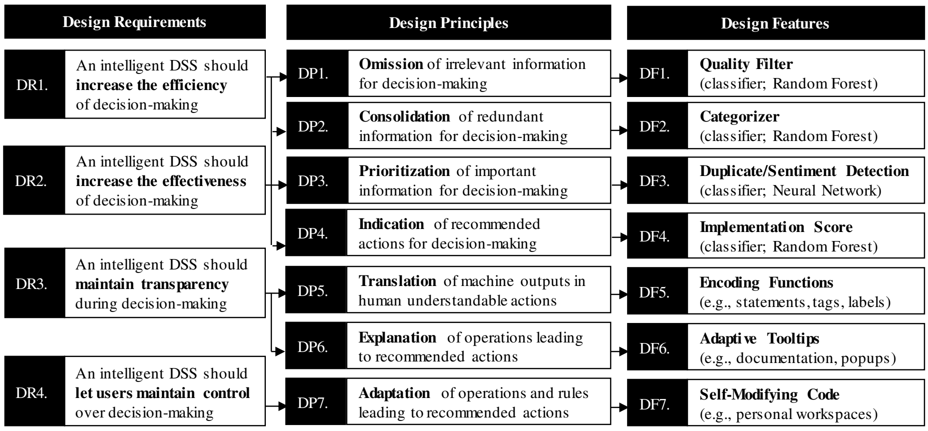

Figure 3. Overview of Design Requirements, Design Principles, and Design Features

For the instantiation of our design principles and design features, we developed a DSS that supports decision-makers (e.g., project managers) in processing and evaluating crowdsourced data (**Figure 4**). Exhibit A shows the system's analytics dashboard [28] that allows decision-makers to visualize key performance indicators and monitor trends. It gives access to aggregated, high-level data and aims to provide a better understanding of the crowdsourced data. The interface is tile-based and offers the decision-makers control over the appearance and the order of the algorithms' results. The underlying functions are explained in tooltips. Exhibit B shows a drill-down view into low-level data. These views are accessible through the dashboard and allow decision-makers to search or scan for specific information in crowdsourced data once interesting patterns or trends have been identified. The DSS prioritizes important contributions, collapses duplicates, and offers recommendations in this view. Based on the decision-makers' actions, verified labels are generated to improve the models in the backend.

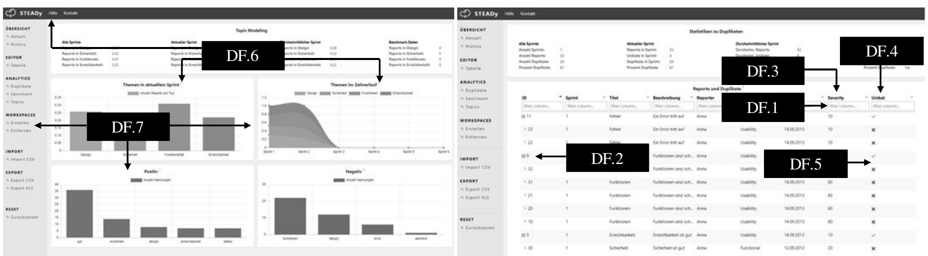

Figure 4. Frontend of the DSS with Design Features (Left: Exhibit A; Right: Exhibit B)

https://doi.org/10.30844/wi_2020_a4-rhyn

4.2 Evaluation

For the evaluation (see **Table 1**), we followed the framework proposed by Venable et al. [36]. This evaluation framework creates a bridge between evaluation goals (*formative* or *summative*) and evaluation strategies (*artificial* or *naturalistic*) in DSR.

Table 1. Overview of Evaluation

	Iteration 1	*Iteration 2*	*Iteration 3*
Goal	Relevancy (formative)	Feasibility (formative)	Usefulness (summative)
Strategy	Qualitative (artificial)	Quantitative (artificial)	Mixed (naturalistic)
Method	Interviews	Machine Learning	Field Tests
Data	31 interviews	300’000 texts	Usage data; 10 interviews

First, we aimed to evaluate whether the requirements, principles, and features are relevant for DSS designs or whether they need to be adapted. For this purpose, we conducted a total of 31 semi-structured interviews with independent subject-matter experts from 20 different organizations (i.e., 5 project managers, 3 executives, 4 heads of departments, 10 QA and test managers, 2 analysts, 3 consultants, 4 innovation managers that are all involved in IT management or data analytics). We asked them to the explain their processes of collecting and analyzing data, describe common problems, and outline potential decision support mechanisms. The results of the interviews confirm that decision-makers are often looking for ways to increase their efficiency (e.g., “*faster reaction times”;* Test Manager, Financial Services; DR1, DP1-4) and effectiveness (e.g., “*categorize feedback to examine the effectiveness of new app releases and updates*”; Project Manager, Analytics Provider; DR2, DP1-4). They also emphasized that transparency (e.g., “*transparent and comprehensive results*”, Innovation Manager, Marketing; DR3, DP5-6) and control (e.g., a “*human-centered approach”* with as much user control as possible; Project Manager, Energy Provider; DR4, DP7) would be imperative for any intelligent DSS used in their jobs. Thus, we find support that our requirements, principles, and features capture relevant components for intelligent DSS.

Table 2. Results of the Formative Evaluation in Iteration 2

	Accuracy		*Additional Measures*	
	Requirement	*Actual*	*Sensitivity*	*Specificity*
DF1: Random Forest	0.75	0.76	0.75	0.77
DF2: Random Forest	0.75	0.86	0.76	0.87
DF3: Cos. Similarity/Neural Net	0.75	0.95	0.83	0.95
DF4: Random Forest	0.75	0.72	0.76	0.69

Second, we aimed to ensure that the principles and features are technically feasible and can be instantiated in prototypes. This can be done by training and testing algorithms with labelled data [29]. In this way, it is possible to evaluate in a controlled

https://doi.org/10.30844/wi_2020_a4-rhyn

setting whether the algorithms are able to technically achieve sufficient performance in terms of accuracy, sensitivity, and specificity to support decision-makers. We used the standard k-fold cross-validation approach with 5 folds or hold-out samples. The results of the training and testing process with data from 676 crowdsourcing projects are listed in **Table 2**. The performance measures show that classification algorithms (DF1-DF4) are capable of achieving the decision-makers' minimum requirements of 75% in terms of accuracy, with a sufficiently high sensitivity and specificity. The implementation of DF5-7 is possible with standard web-technologies for frontend design (e.g., HTML, CSS, Javascript). Hence, all our design principles and features are technically feasible and can be instantiated in a DSS.

As a final, summative evaluation to ensure that our principles and features are useful and address the requirements with a DSS, we conducted 2 field tests with the DSS in organizations. Since we aimed for a naturalistic evaluation [36], we asked the decision- makers to process crowdsourced data (as they would normally do) but use the DSS instead of their standard tools and approaches (e.g., Excel). We collected usage data and interviewed decision-makers to examine the usefulness of the DSS. To evaluate the efficiency (DR1) and the effectiveness (DR2), we followed Sproles [41]. We were interested in the time reduction (in working hours) to process crowdsourced data (i.e., "how well the solution does what it actually does" [41]) and asked the decision-makers whether the DSS supports their tasks (i.e., "the capability of a solution to meet the needs of a problem" [41]). Reviewing crowdsourced data without our DSS was reported to take around 8 hours for 221 contributions. The same process with the DSS takes 4 hours (-50%). In the second case, the reduction was reported to amount to -20%. The decision-makers stated that the reduction is substantial. They explained that DP2 and DP3 are particularly effective to support the process. To investigate transparency (DR3) and decision control (DR4), we interviewed the decision-makers that used the DSS. Following Siau & Wang [39], we aimed to ensure that our design principles and features help decision-makers understand how the DSS is "programmed and what function will be performed in certain conditions". Following Tyler [40], we also aimed to ensure that they grant decision-makers sufficient decision control and "actual influence over the nature of the decision made". In the interviews, the decision-makers confirmed that it is possible for them to understand the DSS's operations and interpret the reliability of the results (DR3) if they are provided with tooltips (DP6) and translated labels (DP5) that explain them. They perceive such systems as transparent. The decision-makers also explained that it is sufficient for them to change parameters (e.g., thresholds; DP7) and have the authority over the final decisions in order to feel in control of the DSS.

5 Discussion

Taken together, the results of our study offer a number of important insights for the design of intelligent DSS in crowdsourcing. A key insight from the first iteration of our DSR study is that the traditional efficiency-effectiveness framework as discussed in existing decision support research [13–15] does not sufficiently capture the deci-

https://doi.org/10.30844/wi_2020_a4-rhyn

sion-makers' requirements for intelligent systems. Surprisingly, however, it is not primarily *trust* that is important for decision-makers as discussed in much related literature [e.g., 39, 42]. Trust is generally defined as "the willingness of a party to be vulnerable to the actions of another party […] irrespective of the ability to *monitor or control* that other party" [43]. However, our results show that decision-makers do want to monitor and control intelligent systems. We find *transparency* and *control* to be crucial for the willingness of decision-makers to work with intelligent DSS and rely on their results. They might also serve as crucial antecedents for trust in intelligent DSS [cf. 39]. Thus, we argue that maintaining transparency and control should be regarded as important meta-requirements for the design of intelligent DSS in crowdsourcing.

Second, insights from our study allow us to understand and explain the mechanisms through which efficiency, effectiveness, transparency, and control can be addressed. We find that, in crowdsourcing, increased efficiency and effectiveness can be explained by a reduction of both manual effort and information load in processing highly unstructured data (DP1-4) [26, 27]. However, even if the DSS is able to efficiently and effectively automate tasks, it is still important to give the users of the system a form of "decision control" [40]. Here, we find that transparency and control are related to (and achievable by) an understanding of the system's functions and an adequate representation of its results (DP5-7) [39]. Our study provides exemplary features for DSS designs.

Third, insights from our study demonstrate that the instantiation of these design principles in intelligent DSS is both technically feasible and economically viable to support decision-makers in crowdsourcing. From a technical perspective, our prototypes show that, even with traditional machine learning approaches (e.g., a random forest algorithm [37]), it is possible achieve performance measures that are sufficient for practical use in crowdsourcing. From an economical perspective, our implemented DSS was able to reduce the manual workload of decision-makers by up to 50%, which may lead to considerable savings in terms of cost and time. In line with existing research [44], we see organizations greatly benefitting from these technologies in crowdsourcing.

5.1 Theoretical Contributions

For research on decision support, we introduce transparency and control as two additional meta-requirements for intelligent systems. Existing decision support research has mostly focused on the traditional efficiency-effectiveness framework [e.g., 13–15]. Recently, however, scholars have emphasized that the growing shift towards large-scale, unstructured data collected from crowds "necessitates reconsidering guidelines for the design product and design process" [8] of DSS. Our study addresses this call and shows that increased efficiency and effectiveness are not sufficient for decision-makers working with intelligent DSS. Instead, we find that transparency and control serve as key components for the adoption of intelligent DSS. While extant research has already discussed requirements that are potentially important for the design of such systems, including trust or explainability [cf. 30], we validate two of

https://doi.org/10.30844/wi_2020_a4-rhyn

them and offer an integrated set of design requirements with corresponding design principles and design features.

For research on crowdsourcing, we capture the theoretical design knowledge for instantiating DSS based on text mining and machine learning. Prior studies have already examined isolated instantiations of these technologies and exemplified their capabilities in domain-specific applications [e.g., 3, 5, 6]. However, research had not yet outlined how to design related systems. Rzepka and Berger [30], for example, explicitly ask how "transparency [can] be ensured for systems that increasingly act autonomously and learn based on machine learning techniques". We extend these studies and provide a set of design principles that guide the deployment and adoption of text mining and machine learning in intelligent DSS. The design principles represent the link between overarching design requirements and concrete design features. They explain how efficiency, effectiveness, transparency, and control in decision-making can be increased.

5.2 Practical Contributions

Our study also offers a number of practical contributions for developers of DSS and managers of crowdsourcing initiatives or platforms. For developers, we describe specific design features (e.g., sentiment analysis, duplicate recognition) that show how the design principles can be instantiated in order to meet the requirements of intelligent DSS in crowdsourcing. For many of these features, we relied on traditional ensemble learning methods (e.g., a random forest algorithm [37]) instead of more complex neural networks. We found the former to achieve sufficient performance for practical use in crowdsourcing. Thus, for better transparency and easier communication of the results, we recommend developers to resort to simpler models whenever possible. Furthermore, we urge developers to not neglect documentation and make use of tooltips that explain the functionality of the algorithms. Even small changes in the interface, such as providing variable importance plots or exposing the algorithm's parameters, greatly benefit the perceived transparency and control when working with DSS.

Second, for managers of crowdsourcing initiatives or platforms, our findings show that intelligent DSS may drastically increase the efficiency and effectiveness of the evaluation of user-generated contributions by crowds. Abbasi et al. [8], for example, emphasize that "IS research needs to not only contribute to the design but also examine the feasibility and effectiveness of such IT artifacts for different stakeholders". Decision-makers that used our DSS were able to reduce the required time to process the contributions by 50%. Based on our findings, we recommend managers of crowdsourcing initiatives to make use of such systems and implement text mining and machine learning algorithms on crowdsourcing platforms. We found that these technologies are both technically feasible and economically viable to support decision-makers in evaluating large amounts of crowdsourced contributions.

https://doi.org/10.30844/wi_2020_a4-rhyn

5.3 Limitations and Future Research

As with all research, there are limitations to the findings presented in this study. First, our study focused specifically on intelligent DSS that deal with textual data in CST. While we believe that our design requirements also translate to other intelligent DSS beyond crowdsourcing, we cannot claim that our design principles and features capture universal design knowledge for all other forms of intelligent DSS. We urge future research to investigate design principles and features for other contexts and examine similarities or differences between them (e.g., study the mutability of our principles [35]).

Second, we followed Meth et al. [32] and captured design knowledge in the form of design requirements, design principles, and design features. We acknowledge that the conceptualization of design requirements, design principles, and design features represents only the first step toward a more comprehensive understanding of IS designs for intelligent DSS. We see great potential in future research to extend our study and delve deeper into principles of implementations for DSS (e.g., methods or processes for organizational adoption [35]) and use patterns (e.g., testable propositions [35]).

Third, we focused on system design rather than system use of intelligent DSS. Thus, an interesting avenue for future research is to study in more detail how different designs of intelligent DSS affect performance in organizations and how decision-makers work with these systems. We strongly believe that these technologies will represent fundamental components for future DSS designs and thus justify further research.

6 Conclusion

In crowdsourcing, it represents a challenge to process textual contributions. Research already examined the technical capabilities of text mining and machine learning to support decision-makers. Yet, it remained unclear how to design intelligent DSS based on these algorithms. We addressed this gap with a DSR approach and developed design requirements, design principles, and design features to guide the development of intelligent DSS in crowdsourcing. Our study shows that intelligent DSS based on these principles are feasible to support decision-makers in evaluating crowdsourced data.

References

1. Blohm, I., Leimeister, J.M., Krcmar, H.: Crowdsourcing: How to Benefit from (Too) Many Great Ideas. MIS Q. Exec. 12, 199–211 (2013).
2. Leicht, N., Rhyn, M., Hansbauer, G.: Can Laymen Outperform Experts? The Effects of User Expertise and Task Design in Crowdsourced Software Testing. In: Proceedings of the

https://doi.org/10.30844/wi_2020_a4-rhyn

24th European Conference on Information Systems (ECIS). pp. 1–11. AIS, Istanbul, Turkey (2016).
3. Barbier, G., Zafarani, R., Gao, H., Fung, G., Liu, H.: Maximizing Benefits from Crowdsourced Data. Comput. Math. Organ. Theory. 18, 257–279 (2012).
4. Piezunka, H., Dahlander, L.: Distant Search, Narrow Attention: How Crowding Alters Organizations' Filtering of Suggestions in Crowdsourcing. Acad. Manag. J. 58, 856–880 (2015).
5. Walter, T.P., Back, A.: A Text Mining Approach to Evaluate Submissions to Crowdsourcing Contests. In: Proceedings of the 46th Hawaii International Conference on System Sciences (HICSS). pp. 3109–3118. IEEE, Waikoloa, Hawaii (2013).
6. Feng, Y., Chen, Z., Jones, J.A., Fang, C., Xu, B.: Test Report Prioritization to Assist Crowdsourced Testing. In: Proceedings of the 10th Joint Meeting on Foundations of Software Engineering (ESEC/FSE). pp. 225–236. ACM, Lombardy (2015).
7. Zhao, Y., Zhu, Q.: Evaluation on Crowdsourcing Research: Current Status and Future Direction. Inf. Syst. Front. 16, 417–434 (2014).
8. Abbasi, A., Sarker, S., Chiang, R.H.L.: Big Data Research in Information Systems: Toward an Inclusive Research Agenda. J. Assoc. Inf. Syst. 17, 1–32 (2016).
9. Wang, W., Benbasat, I.: Trust in and Adoption of Online Recommendation Agents. J. Assoc. Inf. Syst. 6, 72–101 (2005).
10. Chandra, L., Seidel, S., Gregor, S.: Prescriptive Knowledge in IS Research: Conceptualizing Design Principles in Terms of Materiality, Action, and Boundary Conditions. In: Proceedings of the 48th Hawaii International Conference on System Sciences (HICSS). pp. 4039–4048. IEEE, Hawaii (2015).
11. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. J. Manag. Inf. Syst. 24, 45–77 (2007).
12. Österle, H., Otto, B.: Consortium Research. Bus. Inf. Syst. Eng. 2, 283–293 (2010).
13. Todd, P., Benbasat, I.: Evaluating the Impact of DSS, Cognitive Effort, and Incentives on Strategy Selection. Inf. Syst. Res. 10, 356–374 (1999).
14. Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R., Carlsson, C.: Past, Present, and Future of Decision Support Technology. Decis. Support Syst. 33, 111–126 (2002).
15. Wang, W., Benbasat, I.: Interactive Decision Aids for Consumer Decision Making in E-Commerce: The Influence of Perceived Strategy Restrictiveness. MIS Q. 33, 293–320 (2009).
16. Geiger, D., Schader, M.: Personalized Task Recommendation in Crowdsourcing Information Systems - Current State of the Art. Decis. Support Syst. 65, 3–16 (2014).
17. Blohm, I., Zogaj, S., Bretschneider, U., Leimeister, J.M.: How to Manage Crowdsourcing Platforms Effectively? Calif. Manage. Rev. 60, 122–149 (2018).
18. Tushman, M.L.: Special Boundary Roles in the Innovation Process. Adm. Sci. Q. 22, 587–605 (1977).
19. Arnott, D., Pervan, G.: A Critical Analysis of Decision Support Systems Research Revisited: The Rise of Design Science. J. Inf. Technol. 29, 269–293 (2014).
20. Simon, H.A.: The New Science of Management Decision. Prentice Hall, New Jersey (1960).
21. Chiu, C.M., Liang, T.P., Turban, E.: What Can Crowdsourcing Do for Decision Support? Decis. Support Syst. 65, 40–49 (2014).
22. Eppler, M.J., Mengis, J.: The Concept of Information Overload: A Review of Literature from Organization Science, Accounting, Marketing, MIS, and Related Disciplines. Inf. Soc. 20, 325–344 (2004).

https://doi.org/10.30844/wi_2020_a4-rhyn

23. Jacoby, J.: Information Load and Decision Quality: Some Contested Issues. J. Mark. Res. 14, 569–573 (1977).
24. Schick, A.G., Gordon, L.A., Haka, S.: Information Overload: A Temporal Approach. Accounting, Organ. Soc. 15, 199–220 (1990).
25. Swain, M.R., Haka, S.F.: Effects of Information Load on Capital Budgeting Decisions. Behav. Res. Account. 12, 171–198 (2000).
26. Häubl, G., Trifts, V.: Consumer Decision Making in Online Shopping Environments: The Effects of Interactive Decision Aids. Mark. Sci. 19, 4–21 (2000).
27. Silver, M.S.: Decisional Guidance for Computer-Based Decision Support. MIS Q. 15, 105–122 (1991).
28. Holsapple, C., Lee-Post, A., Pakath, R.: A Unified Foundation for Business Analytics. Decis. Support Syst. 64, 130–141 (2014).
29. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, Cambridge (2007).
30. Rzepka, C., Berger, B.: User Interaction with AI-Enabled Systems: A Systematic Review of IS Research. In: Proceedings of the 39th International Conference on Information Systems (ICIS). pp. 1–17. AIS, San Francisco (2018).
31. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Q. 28, 75–105 (2004).
32. Meth, H., Mueller, B., Maedche, A.: Designing a Requirement Mining System. J. Assoc. Inf. Syst. 16, 799–837 (2015).
33. Walls, J.G., Widmeyer, G.R., El Sawy, O.A.: Building an Information System Design Theory for Vigilant EIS. Inf. Syst. Res. 3, 36–59 (1992).
34. Sein, M.K., Henfridsson, O., Rossi, M., Lindgren, R.: Action Design Research. MIS Q. 35, 37–56 (2011).
35. Gregor, S., Jones, D.: The Anatomy of a Design Theory. J. Assoc. Inf. Syst. 8, 312–335 (2007).
36. Venable, J., Pries-Heje, J., Baskerville, R.: FEDS: A Framework for Evaluation in Design Science Research. Eur. J. Inf. Syst. 25, 77–89 (2016).
37. Breiman, L.: Random Forests. Mach. Learn. 45, 5–32 (2001).
38. Theodorou, A., Wortham, R.H., Bryson, J.J.: Designing and Implementing Transparency for Real Time Inspection of Autonomous Robots. Conn. Sci. 29, 230–241 (2017).
39. Siau, K., Wang, W.: Building Trust in Artificial Intelligence, Machine Learning, and Robotics. Cut. Bus. Technol. J. 31, 47–53 (2018).
40. Tyler, T.R., Rasinski, K.A., Spodick, N.: Influence of Voice on Satisfaction With Leaders: Exploring the Meaning of Process Control. J. Pers. Soc. Psychol. 48, 72–81 (1985).
41. Sproles, N.: The Difficult Problem of Establishing Measures of Effectiveness for Command and Control: A Systems Engineering Perspective. Syst. Eng. 4, 145–155 (2001).
42. Wang, L., Jamieson, G.A., Hollands, J.G.: Trust and Reliance on an Automated Combat Identification System. Hum. Factors. 51, 281–291 (2009).
43. Mayer, R.C., Davis, J.H., Schoorman, F.. D.: An Integrated Model of Organizational Trust. Acad. Manag. Rev. 20, 709–734 (1995).
44. Chen, H., Chaing, R.H.L., Storey, V.C.: Business Intelligence and Analytics: From Big Data to Big Impact. MIS Q. 36, 1165–1188 (2012).

https://doi.org/10.30844/wi_2020_a4-rhyn