A Means to what End? Evaluating the Explainability of Software Systems using Goal-Oriented Heuristics

Hannah Deters
Leibniz University Hannover
Hannover, Germany
hannah.deters@inf.uni-hannover.de

Jakob Droste
Leibniz University Hannover
Hannover, Germany
jakob.droste@inf.uni-hannover.de

Kurt Schneider
Leibniz University Hannover
Hannover, Germany
kurt.schneider@inf.uni-hannover.de

ABSTRACT

Explainability is an emerging quality aspect of software systems. Explanations offer a solution approach for achieving a variety of quality goals, such as transparency and user satisfaction. Therefore, explainability should be considered a means to an end. The evaluation of quality aspects is essential for successful software development. Evaluating explainability allows an assessment of the quality of explanations and enables the comparison of different explanation variants. As the evaluation depends on what quality goals the explanations are supposed to achieve, evaluating explainability is non-trivial. To address this problem, we combine the already well-established method of expert evaluation with goal-oriented heuristics. Goal-oriented heuristics are heuristics that are grouped with respect to the goals that the explanations are meant to achieve. By establishing appropriate goal-oriented heuristics, software engineers are enabled to evaluate explanations and identify problems with affordable resources. To show that this way of evaluating explainability is suitable, we conducted an interactive user study, using a high-fidelity software prototype. The results suggest that the alignment of heuristics with specific goals can enable an effective assessment of explainability.

CCS CONCEPTS

• **Human-centered computing** → **Heuristic evaluations**; • **Software and its engineering** → *Software design engineering*; *Requirements analysis.*

KEYWORDS

Explainability, Metrics, Software Evaluation, Heuristics, Explainable Systems

ACM Reference Format:
Hannah Deters, Jakob Droste, and Kurt Schneider. 2023. A Means to what End? Evaluating the Explainability of Software Systems using Goal-Oriented Heuristics. In *Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE '23), June 14–16, 2023, Oulu, Finland.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3593434.3593444

EASE '23, June 14–16, 2023, Oulu, Finland
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0044-6/23/06.
https://doi.org/10.1145/3593434.3593444

1 INTRODUCTION

The evaluation of quality aspects is a major factor in the field of software engineering. Evaluation methods can be used to verify whether the predefined quality requirements have been met. In addition, metrics enable the tracking of improvements in quality. Furthermore, the comparison of two systems with respect to a specific quality aspect are enabled. Due to the importance of this topic, there are already several established methods that support the evaluation of many quality aspects [18, 19]. However, with the emergence of explainability as a new quality aspect, the research community is faced with a new challenge. As explainability can be expected to gain more ground in the industry of the future, it is increasingly important to find suitable ways to evaluate explanations, in order to enable a successful software engineering process.

Explainability has a special nature compared to other NFRs (non-functional requirements). Explanations serve as a means to achieve a variety of quality goals [5]. They are required, for example, so that users have more trust in the system, enjoy using it more or have better insight into the inner workings of the system. This particular property of explainability needs to be taken into account when attempting to evaluate it as an NFR.

The majority of research works concerning explainability focus on the way in which explanations should be provided [23]. This covers not only the contents of the explanations, but also different presentation forms, such as textual explanations or visual examples [1]. In contrast, this work will focus on the evaluation of explainability. We do not intend to discuss how to design explanations that are the most suitable for a certain scenario or user group. Instead, we will investigate ways to evaluate the explainability of already existing software systems, with respect to the intended goals of the explanations.

We present the concept of *goal-oriented heuristics*, and examine their feasibility for evaluating explainability. *Goal-oriented* means that heuristics are grouped based on the goals that are supposed to be achieved by the explanations, so that the evaluation can be done in accordance with these goals. The intended goals of the explanations are defined by the group of people who commission the software system. This group of people is usually represented by the product owner. We focus on the goals of the product owner, as these are the goals that software engineers are working towards. By focusing on these goals, we ensure that the evaluation of explainability determines whether the explanations accomplish what the product owner intended them to do.

2 BACKGROUND AND RELATED WORK

2.1 Explainability

Over the last decades, there has been a plethora of related work on explainability as a non-functional requirement, especially in the context of explainable artificial intelligence [5, 23]. Chazette et al. [5] performed a thorough systematic literature review on the topic. They found that the explainability of a software system depends on the addressee of the explanations and context of use, but also on the explainer, i.e., the entity that provides the explanation. Other works have argued for the importance of the intent or goals of an explanation [9, 13, 35]. Based on this related work, we define explainability:

> **Definition:** Explainability is the ability of a software to be explained to an addressee, given a specific context of use and depending on the goals of the explainer.

In most scenarios, the explainer is the software system itself, providing explanations via its interface to explain itself to the user. We call these systems *self-explaining systems*. In a case like that, the system does not have its own intent or goal when providing the explanations. Rather, it projects the goals of the product owner, who provided their explainability goals when raising the explainability requirements to the software.

Currently, research on explainability is often focusing on the field of AI. Machine learning models are made explainable to enable the user to build more trust in the system. For example, the explainability technique LIME "explains the predictions of any classifier in an interpretable and faithful manner" [27]. In this paper, however, we do not focus on the evaluation of such automatically generated explanations in the domain of XAI. We want to evaluate the explainability of a system on a higher level. This means that we do not want to gauge the ability of an algorithm to provide the correct explanations. Instead, we want to evaluate whether the presented information is appropriate with respect to the end-users and the explanations' intend.

Contemporary research has found that different users have different explainability needs [5, 25, 32]. Supporting those needs, various works argue that explanations should be personalized with regard to their addressee [29, 34]. We aim to extend this concept of personalization towards the addressee by also incorporating the goals of the explainer. In other words, explanations would be provided with the goals of the explainer in mind and would in turn be evaluated with respect to those goals.

Explainability interacts with a variety of other quality aspects such as understandability, transparency and user satisfaction [5]. At first glance, explainability is expected to have a positive influence on those three NFRs. However, explainability may also compete with them. Providing too much information within an explanation might impede the user experience. Depending on the explainer's goals, pushing for higher transparency might come at the cost of user satisfaction and decrease understandability [6]. With respect to that, we hold that explanations should be evaluated with those quality goals in mind.

Most related works focus on techniques and tools for providing explanations. In comparison, the number of works that focus on the means to evaluate explainability is very limited [23]. In their systematic literature review, Nunes and Jannach [23] found that the majority of research papers on explainability do not include any kind of empirical evaluation. Furthermore, most works that employ empirical evaluation do so via user studies. A major downside of user studies is that they can be very costly in terms of resources. Heuristics enable expert evaluations in place of user studies, which are much less resource intensive.

2.2 Metrics and Heuristics

A software metric is a measurement performed on a software, resulting in a numerical value that describes to what degree a certain aspect of the software has been fulfilled [12]. The aspect in question has a direct impact on the quality of the software. Following the definition of metrics, our goal-oriented heuristics should produce a numerical value that is interpretable in terms of the degree to which the goals of an explanation have been satisfied.

The aims pursued with a metric should be identified. Metrics can then be developed and evaluated with respect to those aims. The following goals are frequently mentioned in related works:

- Metrics should support the formulation of quality requirements [12].
- Metrics should enable the analysis of deviations between the established quality requirements and the actual quality of the system [10, 12].
- Metrics should pinpoint possible defects [10, 19].

Heuristics are a special form of metrics. Romanycia and Pelletier reviewed many definitions of heuristics and concluded that heuristics have a "rule of thumbishness" and that they "had to be useful but need not guarantee success" [28]. In terms of software metrics, this means that a heuristic is a reasonable, resource-efficient way to get a measurement right with high probability, but there is no guarantee that it will always produce exactly the correct value.

In 1990, Nielsen [22] introduced heuristic evaluation for usability engineering. He proposes this method for the evaluation of the interface design of software systems. Since explanations are usually a part of a software interface, it is conceivable that the concept of heuristic evaluation is also applicable to explanations. Nielsen defines heuristic evaluation as a "method for finding the usability problems in a user interface design" [21]. Thus, according to Nielsen, heuristic evaluation focuses only on finding problems related to a particular aspect. In this work, we use Likert scales to support the application of heuristics. The use of Likert scales allows us to obtain a first assessment of the quality of explainability, in addition to the identification of possible problems regarding the explanations.

In usability engineering, heuristics are usually applied by experts. For our heuristics, someone can be considered an *expert* if they have a basic understanding of explainability and know basic terms and concepts from the IT area. In the context of heuristic evaluation, Nielsen stated that the evaluation becomes much more effective if several evaluators accomplish the evaluation independently of each other [21]. Therefore, this work will focus on heuristic evaluation by multiple evaluators.

3 GOAL-ORIENTED HEURISTICS

Following the findings of related work, it can be seen that the goal pursued by the explanations plays a major role in explainability.

Unlike other NFRs, explanations can and must take many different forms in order to achieve what is required. For example, long detailed explanations are well suited in a context where the software should certainly ensure that the user understands something. On the other hand, in a context in which it is only important that the user understands approximately, but should be able to do so very quickly, long detailed explanations are very obstructive. Accordingly, depending on the goal of the explanation, widely differing properties of the explanations must be considered desirable. Hence, explanations should be developed with respect to the particular goals they are supposed to achieve. The basic idea behind goal-oriented heuristics is to keep these goals in mind during the evaluation of the explainability of a system. To this end, possible evaluation methods must be mapped to the aspects that they are trying to measure, i.e., their goals. Figure 1 demonstrates this concept.

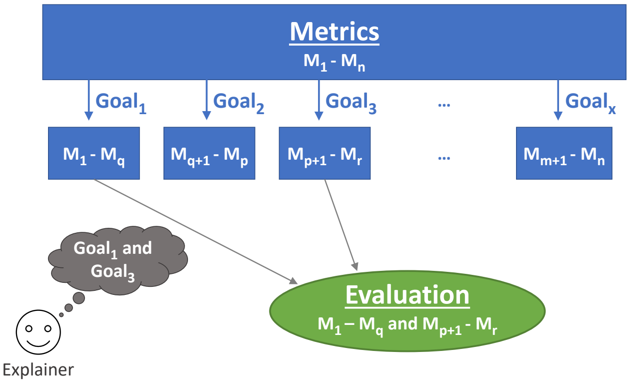


Figure 1: Applying Goal-Oriented Heuristics

The starting point is a set of various metrics for explainability ($M_1 - M_n$). This set of metrics is established in section 3.2, based on our finding from existing literature. In our work, we focus on heuristics only. This initial set of heuristics is then divided into disjoint sets based on possible explainability goals (see section 3.3). In Figure 1, metrics $M_1 - M_q$ were assigned to goal 1, metrics $M_{q+1} - M_p$ were assigned to goal 2 and so on. The explainer in our example seeks goals 1 and 3, meaning that metrics $M_1 - M_q$ and $M_{p+1} - M_r$ should be used.

3.1 Research Design

In order to investigate if the goal-oriented heuristic sets are more effective than the ungrouped heuristic sets, we focus on two objectives that metrics pursue. The first objective is to produce consistent results. In other words, it is desirable that evaluators agree on similar results. Otherwise, the evaluation might be too subjective. The second objective of metrics is to detect differences in quality between systems. If the heuristics are able to detect significant differences between two systems, they are capable of revealing differences in the quality of the explanations. Based on this, we formulate the following research questions:

RQ 1 What are suitable heuristics for the evaluation of explainability?

RQ 2 Do goal-oriented heuristics allow multiple evaluators to better agree on how to evaluate explainability than the ungrouped heuristics?

RQ 3 Are goal-oriented heuristics able to detect more significant differences in terms of explainability than the ungrouped heuristics?

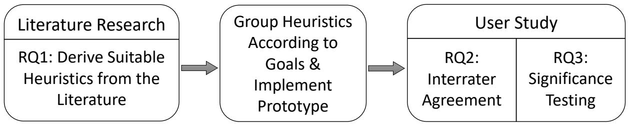

Literature Research
RQ1: Derive Suitable Heuristics from the Literature
Group Heuristics According to Goals & Implement Prototype
User Study
RQ2: Interrater Agreement
RQ3: Significance Testing

Figure 2: Research Design

To answer the research questions, we performed three steps. The first step was to find suitable heuristics to assess the explainability of systems. To obtain a proper foundation, four known papers were used to perform one forward and backward snowballing step to find further papers that refer to the assessment of explainability. Based on these papers, we then developed feasible heuristics. In the second step, the heuristics were grouped with respect to possible goals. Furthermore, a high-fidelity software prototype was implemented to support the subsequent user study. The third step comprises a user study with subsequent data analysis. The interrater agreement is used to answer RQ2 and a significance test is used to answer RQ3. Further details are given in chapter 5.

3.2 Developing Heuristics

Based on our findings from literature, we develop heuristics that on the one hand help to assess the degree of explainability, and on the other hand also pinpoint potential problems that exist regarding explainability. These heuristics can be found in Table 1. It should be noted that the heuristics are not complete, meaning that they do not cover all aspects of explainability. Furthermore, they are not universally applicable - for example, some heuristics can only be applied to textual explanations, but not to visual examples. However, this is not necessary, since our aim is only to show the possibility of evaluating and improving explainability using goal-oriented heuristics.

Heuristic **M1** intends to verify that explanations are as simple as possible. Baaj et al. [2] explicitly state that natural language is desirable for explanations. This underlines the notion that technical terms should also be avoided. The second heuristic is an established method for measuring the complexity of texts. The Flesch Reading Ease score calculates a score based on the number of words, sentences and syllables [38]. Vultureanu-Albişi et al. [36] argue that the number of words and the word length have great influence on the quality of the explanation. Therefore, heuristic **M2** is a good way to assess this matter. Notably, heuristic **M2** requires a calculation with cannot be easily done by hand, so it is rather effortful compared to the other heuristics. Vultureanu-Albişi et al. [36] state that sentences should have a logical relationship to each other. There should be no contradictions between sentences and, if possible, they should follow a common thread, as that increases the understandability of the explanation. This concept is reflected in heuristic **M3**.

Table 1: Heuristics to Assess Explainability

ID	Heuristic	based on
M1	The language is simple – it does not contain any technical words that the target user does not understand.	[2]
M2	Flesch Reading Ease Score (computable score used to assess the readability of texts)	[36, 38]
M3	The elements in the explanation are logically coherent. The explanation follows a common thread and contains no contradictions.	[36]
M4	For each input parameter, it is clear why the system needs the input and what it is used for.	[3, 4, 11]
M5	It is clear how the parameters that the user enters are connected to the events being explained.	[4]
M6	It is clear which aspects the explanation targets.	[16, 17, 26]
M7	The explanation is easy to find.	[8, 17, 20, 33]
M8	The explanation is not disruptive and does not interfere with the general use of the system.	[15, 24, 37]
M9	The explanation is understandable with the prior knowledge of the users of each target group of the system.	[7, 13, 36]
M10	The explanation is adaptable to the users' level of prior knowledge.	[7, 13, 36]

According to Carvalho et al. [4] an explanation should reflect the importance of its own parts and features. Likewise, for each output the system generates, the user should be able to comprehend which input had the greatest influence. Furthermore, Hunt and Price [11] state that an explanation should keep the user informed why certain questions are asked. In other words, it should be clear why certain input parameters are being requested. Especially when it comes to privacy related data, so-called privacy explanations should inform the user what the data will be used for, according to Brunotte et al. [3] Based on these statements, the heuristics **M4** and **M5** were established. The mental model that can be built or improved through explanations is another important property of explainability [16, 17, 26]. A mental model should reflect the real system model as closely as possible. Notably, this criterion is difficult to evaluate to its full extent without user studies. However, starting with a necessary criterion for enabling a correct mental model, we can get a first estimation for this. Explanations need to state which aspects of the system they refer to (**M6**). If this criterion is not fulfilled, the user cannot generate a correct mental model.

Regarding user satisfaction with explanations, some established usability heuristics can be applied to explainability. When custom UI elements like dialogs are created for the explanations, common usability guidelines should be considered. Two aspects that are particularly important are captured in the heuristics **M7** and **M8**. According to Langer et al. [17] explanations should be easy to use. To this end, they should first of all be easy to find. Having the users search for an explanation can be very frustrating for them, and may result in the explanations not being used at all. Furthermore, using the explanations should increase the enjoyment of using the system itself [15, 24, 37]. Therefore, it is crucial for the explanations to not be disruptive and to not interfere with general use (**M8**).

Another important aspect of explanations is the ability to adapt to the user [4, 13, 31]. If a system has diverse target users, possible differences in prior knowledge must be taken into account [7, 13, 36]. In some scenarios, explanations should be adaptable to the user, i.e., to their prior knowledge. This would allow every target user to receive understandable information while avoiding the disturbance of advanced users with unnecessary additional information (**M10**). Either way, it is critical that the explanations shown to the user are comprehensible to them within the scope of their prior knowledge(**M9**).

3.3 Grouping Heuristics

Depending on the purpose of an explanation, there are different properties that are required in order for the explanation to achieve its goals. Consequently, not all explanations can be evaluated according to the same criteria. On one hand, explanations may be designed to help the user understand the inner workings of a system. On the other hand, explanations may aim to increase overall user satisfaction with the system. In such a case, the inner workings of the system would not necessarily be the focus of the explanation. Furthermore, explanations may be used to make the system usable for different user groups. Hence, adapting the explanations to the users' needs is very important. In this section, we group our previously defined heuristics according to the purpose of the explanation. To this end, we establish four categories for heuristics. This classification is shown in Table 2. Within other contexts, these four categories are also known as NFRs. The reason for this is that goals that can be demanded of software are usually requirements that fall into this generally defined set of NFRs. Since explainability refers to software systems, the goals pursued with explanations are often similar to known NFRs. Nevertheless, they are still referred to as goals in order to clearly distinguish them from general requirements.

Table 2: Grouping of the Heuristics

Category	Heuristics
Understandability	M1, M2, M3
Transparency	M4, M5, M6
Satisfaction	M7, M8
Suitability	M9, M10

The first category (*understandability*) contains all heuristics that aim to make explanations as easy to understand as possible. Heuristics M1-M3 focus on the avoidance of technical terms and complicated phrasing, and on ensuring the consistency within the explanation. These heuristics should be used if the purpose of an explanation is to minimize the mental effort required from the user while using the system.

The *transparency* category focuses on ensuring that the inner workings of the system are evident to the user. Heuristics M4-M6 are used to check whether the role and use of the input parameters are clarified, and whether it is comprehensible which aspects exactly are targeted by the explanation. Thus, the associated heuristics should be evaluated if explainability is introduced with the goal of revealing the inner workings of the system.

Heuristics M7 and M8 were combined into the *satisfaction* category. This category should be considered if the purpose of the explanations is to increase the overall satisfaction with the system. Therefore, the heuristics that have been assigned to this category ensure that the explanations are easy to use - more precisely, that the explanations are easy to find and non-disruptive.

Finally, the last two heuristics M9 and M10 were merged into the category *suitability*. These heuristics should be taken into account if the adaptability to users and scenarios is a focus of the system. In this case, they primarily ensure adaptation to the user, or more precisely, to the user's prior knowledge.

Before evaluating explainability using goal-oriented heuristics, one must determine which goals the explanations are supposed to achieve. These goals are usually already clarified during the requirements elicitation process.Thus, they are already determined before development and evaluation are started. Therefore, the first step of specifying the goals of the explanations does not require any additional effort.

4 EVALUATION

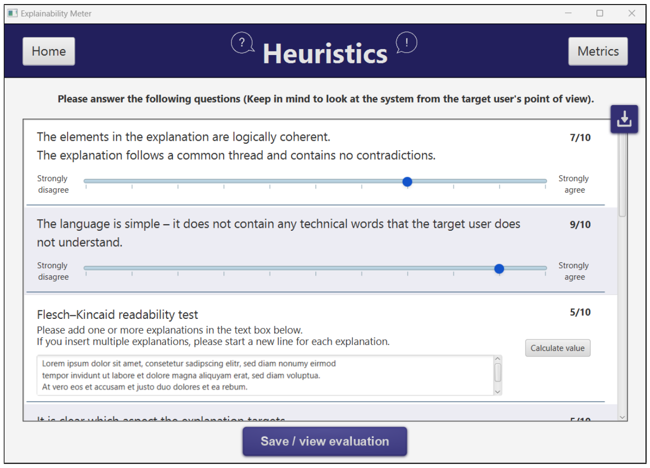
A user study was conducted to analyze the evaluation of explainability using goal-oriented heuristics. Particular attention was paid to the characteristics of good metrics mentioned in section 2.2.

4.1 Evaluation Prototype

In preparation for the user study, we developed a prototype that supports the application of the heuristics.[1] For heuristic M2, the Flesch Reading Ease Score is calculated automatically as soon as a textual explanation is inserted. The remaining heuristics had to be scored by the participants. The heuristics were implemented using discrete sliders so that they can be scored using a Likert scale. Figure 3 shows a screenshot of the prototype where this evaluation was performed. In addition, the prototype provides a feature to display possible problems that were identified during the evaluation. This feature ensures the goal of *pinpointing possible defects for quality improvement* as mentioned in section 2.2.

4.2 Research Design

Our research questions require the evaluation of two distinct systems. We chose a within-subjects design to obtain data on how participants evaluate two systems in direct comparison. Therefore, each participant rated both of the two systems. This approach carries the potential risk of a learning bias. To mitigate this bias, we asked half of the participants to start by evaluating system A and the other half to start with system B. Since the study was conducted during the Covid-19 pandemic, some participants did not want to participate on site. The study was therefore partially conducted online using a remote desktop program so that the prototype did not have to be installed on the participants' computer. The remaining participants conducted the study on site, where a research environment consisting of a laptop, a second monitor, a keyboard and a mouse was provided.



Please answer the following questions (Keep in mind to look at the system from the target user's point of view).

The elements in the explanation are logically coherent.
The explanation follows a common thread and contains no contradictions. 7/10

The language is simple – it does not contain any technical words that the target user does not understand. 9/10

Flesch–Kincaid readability test 5/10
Please add one or more explanations in the text box below.
If you insert multiple explanations, please start a new line for each explanation.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod
tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.
At vero eos et accusam et justo duo dolores et ea rebum.

It is clear which aspect the explanation targets 5/10

Figure 3: Heuristic Page

4.2.1 *Research Objects.*

We selected two similar systems for the evaluation. Both systems are online consultants for bicycles that guide the user through questions and suggest suitable bikes depending on the answers. Both systems use explanation similarly as an approach to explain why inputs are needed, what they are used for, or to support correct input. Since our study was conducted with German-speaking participants, both systems were used in German. The systems are from here on referred to as system A and system B. Both systems seemed to have a relatively high degree of explainability at first glance. We chose these systems to obtain as meaningful results from the study as possible. If we had chosen a system with a very high degree of explainability and a system with very poor explainability, the heuristics would most likely produce more distinctive results, but the informative value of these results would be low. We wanted to find out whether the goal-oriented heuristics are able to detect differences that are not obvious at first glance.

4.2.2 *Participants.*

As mentioned above, the heuristics are designed for *experts* with basic knowledge in the field of IT and especially explainability. They include some technical terms, which is why the participants should have an IT background. Thus, participants should be pursuing a degree or work in the field of IT to meet this requirement. In addition, other demographics such as gender should be close to the real IT industry.

We acquired 20 participants for the study. The majority of the participants (85%) are pursuing a degree in the fields of computer science or business informatics. The remaining participants (15%) are employed in the field of IT (IT specialist, system integration, public service). The gender distribution was biased towards the male gender (male: 85%, female: 15%). However, as this is rather common in the IT sector, this distribution is acceptable. The average age was 25.3 years (min: 21 years, max: 30 years, SD: 6.32).

4.2.3 *Procedure.*

At the beginning of the study, a short briefing was given to the participants. For this purpose, they received a

[1] The prototype is available for replication of the study. Please contact us at *hannah.deters@inf.uni-hannover.de* for further information.

document explaining the study procedure and a declaration of data processing. In the next step, we gave a brief introduction to the topic of explainability, focusing on the aspects that were important for our study. During this step, examples of explanations were shown so that the participants understood what to look for when evaluating explainability. Possible questions regarding explainability and the process were clarified, and it was made clear to the participants that they could skip heuristics if they wanted to.

After the introduction, the participants had to apply the heuristics to both systems using the prototype. For this purpose, they were first asked to look at the first system and click through it to get a good overview. Once they felt ready, they could start assigning values to each heuristic. In order to calculate the value for heuristic M2, participants were asked to choose five explanations from the system and insert them into the prototype to retrieve their Reading Ease Score. After assigning values to the heuristics, participants could save the results and then look at possible explainability problems on the evaluation screen. Subsequently, this process was repeated for the second system. At the end, the participants were asked to fill out a post-study questionnaire to collect the demographic data.

4.2.4 Independent and dependent Variables. The independent variables of this study are the two systems that are being evaluated and the heuristics that are used for the evaluation. Since a within-subject design is used, we derived a value for each heuristic per system. Those values - more precisely, the ratings given - are the dependent variables of the study.

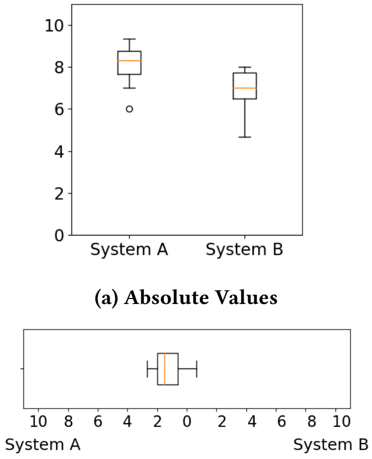
4.3 Data Collection

The study produced two kinds of data. First, for each system, the assigned values per heuristic were stored. These values are natural numbers ranging from zero to ten. This 11-point Likert scale was used to follow Wu and Leung's [40] recommendation, stating that this procedure helps to bring the Likert scale closer to normality and interval scales. For heuristics that the participants did not want to or could not answer, the missing value was replaced with a marker, so that we could later identify them as unanswered. This only occurred once for one heuristic (M7). The values were stored pseudonymously so that we could analyze the data per participant.[2] Second, demographic data was collected from each participant. This included age, occupation and, optionally, field of study.

5 RESULTS

While analyzing the data, we focus on two points that are fundamental for metrics. The first analysis examines the consistency of evaluations across raters. A Metric should, in the best case, always produce the same numerical value regardless of who applies the metric. However, since heuristics do not always produce the exact correct result, it is unlikely that raters will produce the same value for a system per heuristic. Nevertheless, the agreement should be reasonable, otherwise the evaluation will be too unreliable. For this evaluation, we examine the interrater agreement, more precisely the intraclass correlation coefficient.

Second, metrics should be able to capture differences in quality. If a metric always delivers the same value and does not reveal differences in quality, the metric is of no use. Therefore, we investigate whether the heuristics are capable of detecting significant differences between two real systems using significance testing.



(a) Absolute Values

(b) Relative Values

Figure 4: Results of the Understandability Heuristics

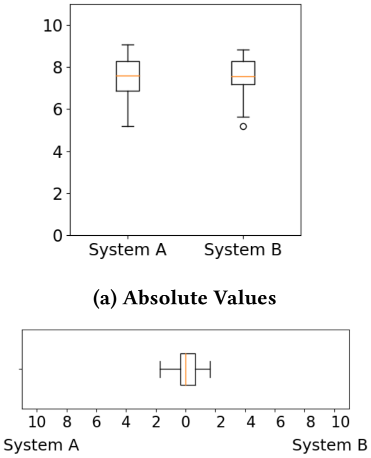
5.1 Data Description

To provide an overview of the collected data, we first present the data using boxplots. For this purpose, the values of the sets were determined using the unweighted average of the corresponding heuristics. For example, for the understandability set, the values of heuristics M1 - M3 were averaged. The values of the ungrouped set correspond to the average of all heuristics (M1 - M10). This way, we can examine whether it is advisable to select metrics that explicitly fit the intended goal or whether all metrics together also achieve a reasonable result.

In addition to the absolute values assigned by each participant for each system, the comparison of the two systems is also of interest. To this end, we need to take a look at the differences between the ratings of the two systems for each participant. These relative values were also presented in a boxplot. The relative values are calculated by subtracting the values of both systems per heuristic for every participant. For example, if a participant rated M1 for system A with 8 points and for system B with 6 points, system A was rated 2 points better. Hence, the relative value of this participant for heuristic M1 is +2 on the side of system A. The final values for the sets are then calculated in the same way as the absolute values (average of the corresponding heuristics).

Figure 4 shows the boxplots for the absolute and relative values of the understandability set. They already suggest that a difference between the systems has been recognized. The boxplot for the absolute values shows that the average of all participants' ratings for system A is over one point higher than for system B. Furthermore, the relative values show that in terms of understandability, system A was rated better than system B by 95% of the participants. The boxplot in Figure 4b shows this distribution. The boxplots of the other sets (transparency, satisfaction, suitability) show similar results. The corresponding figures are provided in appendix A.

[2] The results of the study are available at: https://zenodo.org/record/7872635



(a) Absolute Values

(b) Relative Values

Figure 5: Results of the Ungrouped Heuristics

Figure 5 shows the boxplots for the ratings for all heuristics together, i.e., the ungrouped set. Both boxplots show no noticeable difference between the two systems. The mean of the absolute ratings is almost identical. Similarly, the 75%-quartile of the relative values is just around zero, which means that most participants rated the two systems about the same.

In conclusion, the first visualization of the data already suggests that the goal-oriented heuristics are more capable of detecting differences in quality than the ungrouped set.

5.2 Significance Test

In order to faithfully answer RQ3, we test for statistical significance. More precisely, we use the Mann-Whitney-U test, as the dependent variable is measured on an ordinal scale, but is not normally distributed (thus excluding the independent t-test). In Table 3 the p-values and the average values for system A and system B are given. The first thing to notice is that the ungrouped set does not reveal any significant difference between the two systems. Both systems were evaluated very similarly (system A: 7.46 and system B: 7.5). The p-value further confirms that this difference is not at all statistically significant.

The goal-oriented sets, in contrast, reveal significant differences or in some cases even highly significant differences between the systems. The understandability set shows that system A provides better understandable explanations than system B. According to the p-value of 0.00008, this difference is highly significant. The heuristics of the transparency set (**M4** - **M6**) further show that the explanations from system A also perform better with respect to transparency. Thus, System A's explanations better clarify what the system needs the inputs for and how the system processes the inputs. This difference is statistically significant with a p-value of 0.02. The satisfaction and suitability sets, on the other hand, show that the explanations from system B are better suited to the user group and also increase the overall enjoyment of using the system more than the explanations from system A. Both differences proved to be highly significant with p-values of 0.00131 and 0.00293.

Group	p-value	Average of System A	Average of System B
Ungrouped	**1.0**	7.46	7.5
Understandability	**0.00008**	8.21	6.96
Transparency	**0.02876**	8.15	6.75
Satisfaction	**0.00131**	7.6	9.63
Suitability	**0.00293**	5.2	7.4

Table 3: Significance Values

These results lead to two different conclusions depending on the predefined goal. If the goal of the explanations was to better reveal the inner workings of the system and to convey this information as simply as possible, system A would have a better explainability. In contrast, if the goal of the explanation was to make the system adaptable to different user groups or to increase the overall satisfaction with the system, system B would have better explainability.

Overall, the results of the Mann-Whitney-U test can be summarized as follows: The goal-oriented heuristic sets were able to detect differences in explainability between two systems, whereas the ungrouped set was not able to detect any significant differences. This indicates that grouping metrics according to the goals of the explanations can help to reveal differences in the quality of explainability.

5.3 Interrater Reliability

To answer RQ2, we evaluated the interrater agreement. For this purpose, the intraclass correlation coefficient (ICC) was calculated. Following Koo and Li [14] it "reflects the variation between 2 or more raters who measure the same group of subjects". As the raters are a random sample of the population, two-way random effects occur. Furthermore, according to Shrout and Fleiss [30] the calculation depends on whether the final evaluation is planned to be done by one person or whether the average of several ratings is considered. Based on the recommendation of Nielsen [21], we assume that the heuristic is evaluated by several raters. This assumption leads to the ICC(2,k) defined by Shrout and Fleiss [30]. Nevertheless, we also take a look at the ICC(2,1) to investigate if only one person would also be sufficient for the evaluation.

$$ICC(2,1) = \frac{BMS - EMS}{BMS + (k-1)EMS + k(JMS - EMS)/n} \tag{1}$$

$$ICC(2,k) = \frac{BMS - EMS}{BMS + (JMS - EMS)/n} \tag{2}$$

BMS: Between target variance, JMS: Between judges variance, EMS: Residual variance, n: number of targets, k: number of judges

Table 4 shows the calculated values, including the 95% confidence intervals. The values for ICC(2,1) are below 0.5 for all sets. According to the guideline of Koo and Li [14], this indicates *poor reliability*. Thus, if it is assumed that a single person performs the assessment, the heuristics have poor interrater reliability. In contrast, the values of the ICC(2,k) for the sets understandability, satisfaction and suitability are above 0.9. According to Koo and Li, this indicates *excellent reliability*. Likewise, the result of the transparency set with a

Group	ICC(2,1) | CI95%		ICC(2,k) | CI95%	
Ungrouped	-0.015	[-0.02, 0.0]	-0.436	[-0.46, 0.01]
Understandability	0.492	[0.13, 1.0]	0.95	[0.75, 1.0]
Transparency	0.195	[0.03, 1.0]	0.829	[0.36, 1.0]
Satisfaction	0.345	[0.05, 1.0]	0.913	[0.54, 1.0]
Suitability	0.34	[0.06, 1.0]	0.911	[0.54, 1.0]

Table 4: Intraclass Correlation Coefficient

value of about 0.83 indicates *good reliability*. These findings indicate that the interrater reliability of the goal-oriented heuristics is very good, assuming that several raters are available for the evaluation.

The value of the ICC(2,k) for the ungrouped set, on the other hand, indicates *poor reliability* with a value below 0.5. The reason for this is that the ungrouped set did not find a significant difference between the two systems, as described above. The variance of the measured objects (BMS) plays a major role in the ICC (see equation 1 and 2). This is due to the fact that smaller variations among the raters are not as relevant if the differences between the measured objects are high anyway. Thus, the results again indicate that the goal-oriented heuristics perform better than the ungrouped set.

6 DISCUSSION

6.1 Answering the Research Questions

6.1.1 *RQ1: What are suitable heuristics for the evaluation of explainability?*

The first research question is answered by Table 1. With the help of existing literature, we were able to develop ten heuristics for the evaluation of explainability. Our user study showed that these heuristics enable explainability to be compared between two systems with regard to the quality goals of understandability, transparency, satisfaction, and suitability. As mentioned already, the ten heuristics are not complete, meaning that most likely further heuristics exist. That also applies to the four goals that the heuristics are mapped to.

6.1.2 *RQ2: Do goal-oriented heuristics allow multiple evaluators to better agree on how to evaluate explainability than the ungrouped heuristics?*

In order to answer the second research question, we calculated ICC(2,1) and ICC(2,k). The first conclusion from the results of ICC(2,1) is that the ratings of only one single evaluator have a poor reliability. That means that if only one evaluator is available for evaluation, our heuristics do not produce consistent results. This fact applies to both goal-oriented and ungrouped heuristics. However, examining the ICC(2,k), major differences between the two approaches can be seen. The ungrouped heuristics again produced a value under 0.5 implying a poor reliability. The sets of the goal-oriented heuristics on the other hand produced values over 0.75 and partly even over 0.9 indicating good to excellent reliability. These results strongly indicate that goal-oriented heuristics indeed allow multiple evaluators to better agree on how to evaluate explainability than the ungrouped heuristics.

6.1.3 *RQ3: Are goal-oriented heuristics able to detect more significant differences in terms of explainability than the ungrouped heuristics?*

To further examine whether goal-oriented heuristics are suitable to measure explainability, we investigated if they could detect differences in quality between two systems. More precise, we investigated whether the goal-oriented sets could detect more significant differences than the ungrouped set. To this end, we used the Mann-Whitney-U test. The results show that the ungrouped heuristic set was not able to detect any significant differences. On the other hand, the goal-oriented sets showed statistically significant differences in the quality of the explanations of the two systems. As explanations can pursue many different kinds of goals, these goals can interfere with each another. This coincides with the fact that explainability competes with many other NFRs [5]. In our experiment, the four explanation goals (understandability, transparency, satisfaction and suitability) interacted in such a way that positive and negative properties of the explanations cancelled each other out. Thus, merging the heuristics resulted in the two systems being rated the same, even though there were differences between them. That shows that the evaluation of explanations should be done in accordance with the goals of the explanations, to avoid irrelevant properties from distorting the results. All together, our results suggest that goal-oriented heuristics are more likely to detect differences in quality between two systems.

6.2 Proposed Evaluation Method

Overall, the results of the evaluation indicate that grouping heuristics regarding possible goals, leads to a better reliability of results and allows them to detect more differences in quality. Based on this, we suggest the following procedure while evaluating explainability:

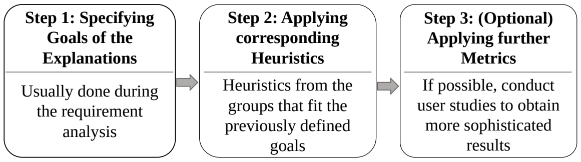

Step 1: Specifying Goals of the Explanations
Usually done during the requirement analysis
Step 2: Applying corresponding Heuristics
Heuristics from the groups that fit the previously defined goals
Step 3: (Optional) Applying further Metrics
If possible, conduct user studies to obtain more sophisticated results

Figure 6: Steps

Our first step is to specify the goals of the explanations. This step is already deeply integrated in the requirements analysis process and is elaborated between the customer who commissioned the software and the company developing it. This means the goal originates from the customer, who consults the requirements engineers. Subsequently, the software is developed towards these goals.

In the second step, the quality of the explainability of the system is assessed. For this purpose, goal-oriented heuristics can be used. In this process, several experts apply the heuristics according to the predefined goals.

If the heuristic evaluation reveals issues that need to be investigated further, an optional third step can be performed. User studies produce more reliable results than heuristics, since heuristics do not guarantee correct results. Thus, if major modifications are considered, it would be reasonable to first ensure that those modifications are necessary. The metrics should also be mapped to explanation goals and applied accordingly.

Based on our results, we believe that our proposed procedure enables a resource-efficient but reliable way to evaluate the explainability of software systems.

6.3 Threats to Validity

To address the validity of our results, we follow the suggestions of Wohlin et al. [39] considering four types of validity. The *construct validity* expresses to what extent the research questions being discussed are represented by the operational measures that are used [39]. The user study that is used to compare the goal-oriented heuristics to the ungrouped heuristics set is a reliable way to measure our research questions. The literature research, on the other hand, carries the risk that we did not find all heuristics that exist for explainability. To this end, a systematic literature review would have been a more reliable method. However, since our goal was not to find all possible heuristics, this threat to validity is not as severe.

The *internal validity* may be threatened by possible biases that influence the reliability of the results. As two systems were evaluated by each participant, a possible learning bias could have occurred. To mitigate this bias, we let one half of the participants begin with system A and the other half with system B. Another threat to the internal validity may be the choice of participants. Since we were not able to acquire real experts for the study, we tried to approximate this expert status with general knowledge in the field of computer science and a brief introduction to the field of explainability to mitigate this threat. Nevertheless, it is possible that real experts would have produced different results if they had used the heuristics.

This choice of systems, on the other hand, increases the *external validity*, as real systems can be generalized better than systems that are developed for a user study only. The external validity could also be threatened by choice of participants, as most of the participants were students, making the results less generalizable. In addition, the generalizability with respect to other possible goals of explanations (except for understandability, transparency, satisfaction and suitability) has to be investigated in future research.

The *conclusion validity* concerns the statistical power of the results and the right application of statistical tests [39]. Since we used well-established statistical methods, the conclusion validity is not threatened.

7 CONCLUSION AND FUTURE WORK

Evaluating explainability is not trivial, as it depends heavily on what the goal of the explanations is. As explainability interacts with many other NFRs and affects some NFRs negatively [5], it must be determined how explanations should be implemented to meet their goals but cause few negative effects. To evaluate explainability in terms of predefined goals, we introduced the concept of goal-oriented heuristics. Different goals of explanations lead to different properties that have to be realized in the explanations. The heuristics are grouped with respect to these properties, so that the heuristics of each set measure whether their particular goal is achieved.

The results of our user study suggest that the goal-oriented sets perform better than the ungrouped set in terms of detecting differences in quality of explainability. The goal-oriented sets were able to reveal significant differences in terms of the quality of explainability between two systems. The ungrouped set, on the other hand, was not able to identify any significant differences. Furthermore, the goal-oriented heuristics produced a higher interrater agreement. Overall, the results suggest that goal-orientation improves the quality of heuristics, in the context of explainability.

The concept of grouping heuristics in accordance with the goals the explanations are supposed to achieve may be transferable to metrics in general. As a consequence, not only heuristics, but any other metrics, should be grouped according to possible goals explanations could achieve.

In future work, we plan to investigate the possible goals to be achieved with explanations. By defining a set of possible goals, we want to enable a consistent grouping of metrics. In addition, we want to compile a useful set of metrics that are mapped to these goals. Additionally, when compiling metrics, further heuristics can be developed which cover further goals of explanations. The collected metrics should then be evaluated in a further user study to examine whether the concept of goal-oriented heuristics is transferable to other metrics and to other goals.

ACKNOWLEDGMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Grant No.: 470146331, project softXplain (2022-2025).

REFERENCES

[1] Amina Adadi and Mohammed Berrada. 2018. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access* 6 (2018), 52138–52160.

[2] Ismaïl Baaj and Jean-Philippe Poli. 2019. Natural Language Generation of Explanations of Fuzzy Inference Decisions. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 1–6. https://doi.org/10.1109/FUZZ-IEEE.2019.8858994

[3] Wasja Brunotte, Alexander Specht, Larissa Chazette, and Kurt Schneider. 2023. Privacy explanations – A means to end-user trust. *Journal of Systems and Software* 195 (2023), 111545. https://doi.org/10.1016/j.jss.2022.111545

[4] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. 2019. Machine Learning Interpretability: A Survey on Methods and Metrics. *Electronics* 8, 8 (2019). https://doi.org/10.3390/electronics8080832

[5] Larissa Chazette, Wasja Brunotte, and Timo Speith. 2021. Exploring explainability: a definition, a model, and a knowledge catalogue. In *2021 IEEE 29th international requirements engineering conference (RE)*. IEEE, 197–208.

[6] Larissa Chazette and Kurt Schneider. 2020. Explainability as a non-functional requirement: challenges and recommendations. *Requirements Engineering* 25, 4 (2020), 493–514.

[7] Li Chen and Pearl Pu. 2005. Trust building in recommender agents. In *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*. Citeseer, 135–145.

[8] M. Sinan Gönül, Dilek Önkal, and Michael Lawrence. 2006. The effects of structural characteristics of explanations on use of a DSS. *Decision Support Systems* 42, 3 (2006), 1481–1493. https://doi.org/10.1016/j.dss.2005.12.003

[9] Robert R Hoffman, Gary Klein, and Shane T Mueller. 2018. Explaining explanation for "explainable AI". In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 62. SAGE Publications Sage CA: Los Angeles, CA, 197–201.

[10] Tu Honglei, Sun Wei, and Zhang Yanan. 2009. The Research on Software Metrics and Software Complexity Metrics. In *2009 International Forum on Computer Science-Technology and Applications*, Vol. 1. IEEE, 131–136. https://doi.org/10.1109/IFCSTA.2009.39

[11] J.E. Hunt and C.J. Price. 1988. Explaining qualitative diagnosis. *Engineering Applications of Artificial Intelligence* 1, 3 (1988), 161–169. https://doi.org/10.1016/0952-1976(88)90002-4

[12] Institute of Electrical and Electronics Engineers. 1993. IEEE Standard for a Software Quality Metrics Methodology. *IEEE Std 1061-1992* (1993), 1–96. https://doi.org/10.1109/IEEESTD.1993.115124

[13] Robert Kass and Tim Finin. 1988. The Need for User Models in Generating Expert System Explanation. *Int. J. Expert Syst.* 1, 4 (1988), 345–375.

[14] Terry K. Koo and Mae Y. Li. 2016. A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research. *Journal of Chiropractic Medicine* 15, 2 (2016), 155–163. https://doi.org/10.1016/j.jcm.2016.02.012

[15] Pigi Kouki, James Schaffer, Jay Pujara, John O'Donovan, and Lise Getoor. 2019. Personalized Explanations for Hybrid Recommender Systems. In *Proceedings of the 24th International Conference on Intelligent User Interfaces* (Marina del Ray, California) *(IUI '19)*. Association for Computing Machinery, New York, NY, USA, 379–390. https://doi.org/10.1145/3301275.3302306

[16] Todd Kulesza, Simone Stumpf, Margaret Burnett, and Irwin Kwan. 2012. Tell Me More? The Effects of Mental Model Soundness on Personalizing an Intelligent Agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/2207676.2207678

[17] Markus Langer, Kevin Baum, Kathrin Hartmann, Stefan Hessel, Timo Speith, and Jonas Wahl. 2021. Explainability Auditing for Intelligent Systems: A Rationale for Multi-Disciplinary Perspectives. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*. IEEE, 164–168. https://doi.org/10.1109/REW53955.2021.00030

[18] Ming-Chang Lee and To Chang. 2013. Software measurement and software metrics in software quality. *International Journal of Software Engineering and Its Applications* 7, 4 (2013), 15–34.

[19] Tsvetelina Mladenova. 2020. Software Quality Metrics – Research, Analysis and Recommendation. In *2020 International Conference Automatics and Informatics (ICAI)*. IEEE, 1–5. https://doi.org/10.1109/ICAI50593.2020.9311361

[20] Sidra Naveed, Tim Donkers, and Jürgen Ziegler. 2018. Argumentation-Based Explanations in Recommender Systems: Conceptual Framework and Empirical Results. In *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization* (Singapore, Singapore) *(UMAP '18)*. Association for Computing Machinery, New York, NY, USA, 293–298. https://doi.org/10.1145/3213586.3225240

[21] Jakob Nielsen. 1995. How to conduct a heuristic evaluation. *Nielsen Norman Group* 1, 1 (1995), 8.

[22] Jakob Nielsen and Rolf Molich. 1990. Heuristic Evaluation of User Interfaces *(CHI '90)*. Association for Computing Machinery, New York, NY, USA, 249–256. https://doi.org/10.1145/97243.97281

[23] Ingrid Nunes and Dietmar Jannach. 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction* 27, 3 (2017), 393–444.

[24] Ingrid Nunes and Dietmar Jannach. 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction* 27, 3 (2017), 393–444.

[25] Alun D. Preece, Dan Harborne, Dave Braines, Richard Tomsett, and Supriyo Chakraborty. 2018. Stakeholders in Explainable AI. *CoRR* abs/1810.00184 (2018). arXiv:1810.00184 http://arxiv.org/abs/1810.00184

[26] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Model-Agnostic Interpretability of Machine Learning. https://doi.org/10.48550/ARXIV.1606.05386

[27] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 1135–1144. https://doi.org/10.1145/2939672.2939778

[28] Marc H. J. Romanycia and Francis Jeffry Pelletier. 1985. What is a heuristic? *Computational Intelligence* 1, 1 (1985), 47–58. https://doi.org/10.1111/j.1467-8640.1985.tb00058.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8640.1985.tb00058.x

[29] Johannes Schneider and Joshua Handali. 2019. Personalized explanation in machine learning. *CoRR* abs/1901.00770 (2019). arXiv:1901.00770 http://arxiv.org/abs/1901.00770

[30] Patrick E Shrout and Joseph L Fleiss. 1979. Intraclass correlations: Uses in assessing rater reliability. *Psychological bulletin* 86, 2 (1979), 420–428.

[31] Kacper Sokol and Peter Flach. 2020. Explainability Fact Sheets: A Framework for Systematic Assessment of Explainable Approaches. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain) *(FAT* '20)*. Association for Computing Machinery, New York, NY, USA, 56–67. https://doi.org/10.1145/3351095.3372870

[32] Kacper Sokol and Peter Flach. 2020. One explanation does not fit all. *KI-Künstliche Intelligenz* 34, 2 (2020), 235–250.

[33] Maxwell Szymanski, Martijn Millecamp, and Katrien Verbert. 2021. Visual, Textual or Hybrid: The Effect of User Expertise on Different Explanations. In *26th International Conference on Intelligent User Interfaces* (College Station, TX, USA) *(IUI '21)*. Association for Computing Machinery, New York, NY, USA, 109–119. https://doi.org/10.1145/3397481.3450662

[34] Nava Tintarev and Judith Masthoff. 2007. Effective explanations of recommendations: user-centered design. In *Proceedings of the 2007 ACM conference on Recommender systems*. 153–156.

[35] Nava Tintarev and Judith Masthoff. 2011. Designing and evaluating explanations for recommender systems. In *Recommender systems handbook*. Springer, 479–510.

[36] Alexandra Vultureanu-Albişi and Costin Bădică. 2021. Recommender Systems: An Explainable AI Perspective. In *2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 1–6. https://doi.org/10.1109/INISTA52262.2021.9548125

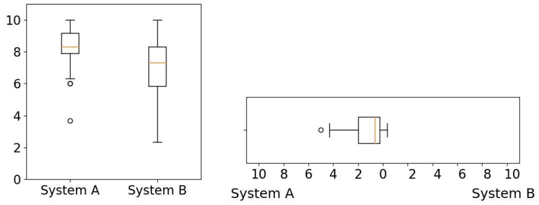
[37] Weiquan Wang, Lingyun Qiu, Dongmin Kim, and Izak Benbasat. 2016. Effects of rational and social appeals of online recommendation agents on cognition- and affect-based trust. *Decision Support Systems* 86 (2016), 48–60. https://doi.org/10.1016/j.dss.2016.03.007

[38] William H. DuBay. 2007. *Unlocking Language: The Classic Studies in Readability.* BookSurge Publishing.

[39] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering.* Springer Science & Business Media.

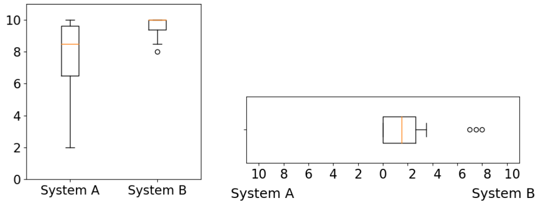
[40] Huiping Wu and Shing-On Leung. 2017. Can Likert scales be treated as interval scales?—A Simulation study. *Journal of Social Service Research* 43, 4 (2017), 527–532.

A GRAPHS



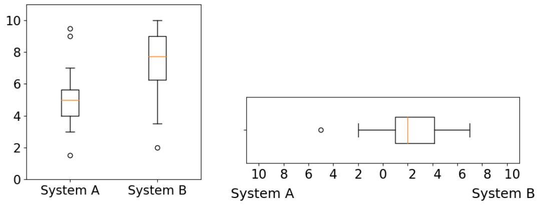
(a) Absolute Values **(b) Difference Values**

Figure 7: Data from Heuristics of the Group Transparency



(a) Absolute Values **(b) Difference Values**

Figure 8: Data from Heuristics of the Group Satisfaction



(a) Absolute Values **(b) Difference Values**

Figure 9: Data from Heuristics of the Group Suitability