How to Support Users in Understanding Intelligent Systems? Structuring the Discussion

MALIN EIBAND, LMU Munich

DANIEL BUSCHEK, Research Group HCI + AI, Department of Computer Science, University of Bayreuth

HEINRICH HUSSMANN, LMU Munich

The opaque nature of many intelligent systems violates established usability principles and thus presents a challenge for human-computer interaction. Research in the field therefore highlights the need for transparency, scrutability, intelligibility, interpretability and explainability, among others. While all of these terms carry a vision of supporting users in understanding intelligent systems, the underlying notions and assumptions about users and their interaction with the system often remain unclear.

We review the literature in HCI through the lens of implied user questions to synthesise a conceptual framework integrating user mindsets, user involvement, and knowledge outcomes to reveal, differentiate and classify current notions in prior work. This framework aims to resolve conceptual ambiguity in the field and enables researchers to clarify their assumptions and become aware of those made in prior work. We thus hope to advance and structure the dialogue in the HCI research community on supporting users in understanding intelligent systems.

To appear in the Proceedings of the 26th ACM Annual Conference on Intelligent User Interfaces (IUI ’21), peer-reviewed author version 5.0, 18. February 2021

CCS Concepts: • Human-centered computing → HCI theory, concepts and models.

Additional Key Words and Phrases: Review, intelligent systems, scrutability, interpretability, transparency, explainability, intelligibility, accountability, interactive machine learning, end-user debugging

ACM Reference Format:

1 INTRODUCTION

Interactive intelligent systems violate core interface design principles such as predictable output and easy error correction [6, 31]. This makes them hard to design, understand, and use [101] – an observation that has already been made decades earlier [46], but it is only in the last years that machine learning has increasingly penetrated everyday applications and thus refueled the discussion on how we want interaction with such systems to be shaped.

One particularly challenging property of intelligent systems is their opaqueness. As a result, researchers [3, 10, 41], practitioners [21], policy-makers [79] and the general public [60] increasingly call for intelligent systems to be transparent [34], scrubtable [55], explainable [81], intelligible [68] and interactive [31], among others, which we will henceforth refer to as system qualities. Work on the system qualities follows a joint and urgent maxim: Designing
interaction in a way that supports users in understanding and dealing with intelligent systems despite their often complex and black-box nature.

Linked by this shared goal, the diverse terms are often employed interchangeably – and yet, prior work implies divergent assumptions about how users may best be supported. For instance, work on interpretability has recently been criticised for unclear use of the term [29, 72], a survey on explainability in recommenders found incompatible existing taxonomies [77], and discussions about system transparency and accountability revealed diverging assumptions (i.e. disclosing source code vs system auditing through experts) [32]. A recent HCI survey shows the fractured terminological landscape in the field [1].

In particular, for supporting user understanding of intelligent systems, clarifying concepts and connecting diverse approaches is crucial to advance scholarship, as pointed out in a recent “roadmap” towards a rigorous science of interpretability [29]. More generally speaking, a lack of conceptual clarity impedes scientific thinking [47] and presents challenging problems for researchers in the respective field: First, a lack of overarching scientific thinking [47] and presents challenging problems for researchers in the respective field: First, a lack of overarching conceptual frameworks renders new ideas difficult to develop and discuss in a structured way. Second, blurred terminological boundaries impede awareness of existing work, for example through varying use of keywords. Third, new prototypes often remain disconnected from the existing body of design solutions. To address this, we need a clearer conceptual understanding of the assumptions that underlie how prior work envisions to foster user understanding of intelligent systems.

In this paper, we thus aim to answer the following research questions:

RQ1: Which assumptions about users and interaction with intelligent systems do researchers make when referring to the system qualities?

RQ2: How can we structure and differentiate these assumptions?

2 CONTRIBUTION AND SUMMARY

We analyse both theoretical concepts and prototype solutions through the lens of implied user questions and synthesise a conceptual framework integrating user mindsets, user involvement and knowledge outcomes to reveal, differentiate and classify notions of supporting user understanding of intelligent systems in prior work.

Our analysis revealed three categories that capture and differentiate current assumptions about users and interaction with intelligent systems from an HCI perspective (also see Figure 1):

(1) User mindsets – what users seek to know (e.g. do they want to know why the system did X, what it was developed for, how trustworthy it is, etc.).
User involvement – how users gain knowledge (e.g. do they actively inquire into the system or do they get presented information by it), and

Knowledge outcomes – what kind of knowledge users gain (e.g. about a specific output or how to correct errors).

In particular, as we will describe later in more detail, we argue that these three categories are linked to users’ intentions when using a system, influence the direction of information transfer between user and system, and reflect the envisioned outcome of the system qualities, respectively.

Our view helps to resolve conceptual ambiguity in the field and provides researchers with a framework to clarify their assumptions and become aware of those made in prior work. We thus hope to advance and structure the dialogue in the HCI research community on supporting users in understanding intelligent systems.

3 SCOPE AND FOUNDATIONS

Before we present the results of our analysis in detail, we first discuss fundamental conceptual prerequisites for our work and locate our own perspective.

3.1 Intelligent Systems

Our work focuses on interaction with intelligent systems. Following Singh [89], a system is intelligent if we need to “attribute cognitive concepts such as intentions and beliefs to it in order to characterize, understand, analyze, or predict its behavior”. While we are aware of the fact that many other definitions of intelligent systems exist, Singh’s definition hints at the potential complexity of intelligent systems and the resulting challenges for users to understand them, and thus motivates work on supporting users in doing so, including this paper.

3.2 A Pragmatic View on Supporting User Understanding

One can identify two opposing perspectives in the larger discussion of supporting users in understanding intelligent systems: A normative and a pragmatic one [33].

The normative perspective is visible in the ethical discourse about intelligent systems or reflected in legislation. It provides users with what has been called a “right to explanation” [39], such as recently articulated in the GDPR [79], and ties lawful use of intelligent systems to the ability to make users understand their decision-making process [44]. While highly valuable for a legal basis for interaction with intelligent systems, this perspective stems from ethical and moral reflection, not from user needs for interaction with a concrete system. As such, it often lacks specifics on how to implement its claims in a way that benefits users in practice.

In this paper, we therefore adopt the pragmatic perspective, which strives to best support users during interaction with intelligent systems. We define the purpose of this support by transferring a statement by Lynham from philosophy of science [73] to the context of our work: Supporting users in understanding intelligent systems means helping people to use a system better and in more informed ways, and to better ends and outcomes.

We argue that this perspective captures well and articulates a core assumption of work on the system qualities: We as HCI researchers in the field of intelligent systems strive to create interfaces and interactions that are explainable, understandable, scrutable, transparent, accountable, intelligible, and so on, precisely because we envision users to then interact with these systems in more informed, effective and efficient ways.
3.3 User Knowledge and Understanding

In general, HCI has widely adopted mental models [53] as representations of the knowledge users possess about a system [76], and this is no different in work on the system qualities (e.g. [33, 62, 95]). Mental models originate from a constructivist perspective on knowledge, where knowledge is seen as individually constructed, subjective interpretations of the world, based on previous experiences and assumptions [97]. In this paper, we adopt this perspective, and use knowledge interchangeably with understanding. Moreover, we assume that knowledge is gained through the transmission of information between user and system.

4 METHOD

Here, we shortly describe the process of paper collection and interpretation through which we derived our framework.

4.1 Theoretical Sampling

Our paper set was collected using theoretical sampling, an approach to collection of qualitative data introduced by Glaser and Strauss as a part of Grounded Theory [37]. In contrast to statistical sampling, where the sample size is fixed a priori, theoretical sampling gradually defines the sample during the interpretation process until theoretical saturation is reached (i.e. the point where further data and interpretation does not further enrich the emerging categories).

We started our sampling by looking for a set of the most widely-adopted system qualities in the field. We did so first through collecting search terms for system qualities based on our experiences as researchers working at the intersection of HCI and AI for several years and then expanded our set of system qualities through the topic networks presented by Abdul et al. [1] in their 2018 survey of over 12,000 papers at this intersection. Their analysis surfaced many system qualities that we sought to address a priori (e.g. interpretability, scrutability, explainability), but also related topics (e.g. accountability and different types of transparency). With keyword searches on the ACM Digital Library, we then iteratively collected papers on these terms, following the above sampling method. We started out with the most cited papers in the field, which we interpreted as described in Section 4.2 to create first categories. Papers were selected to represent the diversity of approaches in the field, but also according to the below criteria:

1. The presented contribution focuses on the system qualities and is linked to intelligent systems, and
2. the contribution involves an HCI perspective (e.g. via a prototype, user study, design guidelines, etc.).

Papers were re-visited if necessary, and we integrated new data through snowball searches as well as through updated keyword searches when we extended our system qualities set. Overall, 222 papers contributed to this process before we considered our categories to be theoretically saturated and had defined our framework dimensions. The final set of system qualities emerging from this process included scrutability, interpretability, transparency, explainability, intelligibility, interactivity (interactive Machine Learning), debuggability (end-user debugging), and accountability.

The paper set used is available on the project website: https://www.medien.ifl.lmu.de/howtosupport/

4.2 Paper Coding and Interpretation

We followed the coding process as suggested by Glaser and Strauss [37] and Strauss and Cobin [90] to code the papers. The coding was done by the first two authors, in joint work on the texts and discussion.

4.2.1 Open Coding: Opening Up the Text. To keep the coding process manageable, we mainly focused on the motivation, contribution and conclusion of a paper, making use of the flexibility of the approach to include other sections as needed
### System qualities | User questions (examples) | User mindsets | User involvement | Knowledge outcomes and qualities
--- | --- | --- | --- | ---
**Accountability** | How fair and controllable is the system? [81] How fair are system decisions? [12] How will algorithmic decision-making impact my (social) work practices? [17] How can I as a designer/developer of decision systems support human values? [49, 98] | Users critique the system, often in a wider context beyond use (e.g. legal, ethical concerns). | Users get informed by the system, or by a third party reporting on the system (e.g. journalists) [25, 27]. People discuss the system and/or challenge its implications beyond use [86], e.g. in its organisational context [17, 45, 96]. | Users seek reflection on outputs, processes, as well as on reasons behind and implications of the system (meta) | **Debuggability** (end-user debugging) | How can I correct system errors? How can I tell the system why it was wrong? [65] How can I give effective feedback to the system? Which objects do I have to change? How do changes affect the rest of the system? What does the system’s feedback mean? How can I detect a system error? How can I find out the cause for an error? How can I solve this error? [66] | Users gain insight into the system to fix its errors. | Users fix the system’s errors. | Users need to understand outputs, processes, and interactions to give good feedback and correct the system. Users make the system more relevant to them by correcting system errors. |
**Explainability** | Can I trust this model? [48] Should I trust this prediction? [42, 83] What are the strengths and limitations of the system? How can I add my knowledge and skills to the decision process? [44] How are input and output related? [52] Why does the system think that I want/need X? [12, 28] Why is this recommendation ranked at the top? [94] | Users gain insight into the system to better use it. | Users get informed by the system. | Users get information about outputs and processes. Explanations should be relevant to the user. They should be “sound and complete”, but not overwhelming [61, 64]. |
**Interactivity (interactive machine learning)** | How can I assess the state of the learned concept? Where does the model fail? Why did the system fail in this specific instance? [31] How well does the system know the domain? How sure is the system that a given output is correct? Did the system do a simple or complex thing to arrive at the output? [86] How to combine models? [91] Which model works best? [7] | Users inspect the system state to refine it or guide its training [31]. | Users iteratively refine the system and guide its training by giving feedback [31]. | Users need to understand outputs, processes, and interactions to guide the system. What is relevant to know is defined by the machine learning task that users and system solve together. |
**Interpretability** | How sensible – and not arbitrary or random – is the system? [4, 81] Why/How questions [56] Can you trust your model? What else can it tell you about the world? [72] | Users gain an axiomatic and interpretive insight into the system to bridge the gap between the system’s criteria and full real-world context [29, 56]. | Users get information about the system’s inner workings [29, 72]. | Users can access information about outputs and processes, which may include low-level (expert) information (e.g. on inner states [56]). What is relevant to know depends on the user’s task with the system. Explicit call for rigorous evaluation [29]. |
**Scrutability** | Why/How did the system do X? What else does the system think I don’t know? What would the system do if I did Y? What does the system do for other people? How can I tell the system what I don’t want? [55] How can I efficiently improve recommendations? [11] | Users want to be able to interpret the system’s decisions. They may analyse and control it for more efficient use. | System decision and behaviour is based on a user model, which users can adequately access and control. Users make “real effort” [55]. | Users gain understanding of outputs and processes. They may also learn about interactions to influence how the system uses the user model. Information should be relevant to users, yet they may also learn about what the system considers relevant. |
**Transparency** | How does the system produce an output (i.e. data sources, reasoning steps)? Why did the system do sth. (i.e. justification, motivation behind the system)? What is informed by the intelligent system (i.e. reveal existence of intelligent processing)? How was the system developed and how is it continually being improved? [41] How did the system produce the model? [36] | Users interpret the system’s output and question the underlying mechanisms. | Users get informed by the system. | Users seek understanding of outputs and processes, also beyond use (meta). What is relevant – and to what extent – may depend on the context of the user’s inquiry. |

Table 1. The system qualities in focus of our work through the lens of our framework, along with example user questions and implied user mindsets, user involvement, and knowledge outcomes. Note that we do not attempt to redefines the terms here but rather provide a guiding overview of our coding, which includes some overlap between terms, as found in the literature.
for a clearer understanding. We first collected open codes about the assumptions on supporting user understanding made in the text. Open coding was guided by so-called basic questions [37] (e.g. Who?, What?, How?, etc.).

This step resulted in a first set of (sub-)categories, namely Motivation (of the work), Key Phenomena and Constructs, Support as Property or Process, Reference to Specific System Part, Main Goal of Support, Main Challenge of Support, User Role, System Role, Interaction Design Implications, and Concrete Realisation.

4.2.2 Axial Coding: Eliciting User Questions. We then created axial codes to refine and differentiate our categories. During this process, we realised that suitable axial codes could be expressed in the form of questions users have about the workings of intelligent systems (e.g. Why did the system do X?). This is in line with prior work: First established by Lim and Dey [69, 71] as a manifestation for users’ information demand, such user questions have gained popularity in related work on the system qualities as a way of anchoring design suggestions and solutions, even if these questions are not always elicited from actual users (e.g., cf. work by Kay and Kummerfeld [55] (scrutability), Kulesza et al. [65] (end-user debugging), or Rader et al. [81] (transparency/accountability)).

In our case, coding these questions helped us to extract and contrast underlying perspectives: On the one hand, they connected prior work on the system qualities, on the other they revealed conceptual differences, and thus refined our preliminary categories. During this process, we kept close to the text of a paper. For example, “Accountability [...] the extent to which participants think the system is fair and they can control the outputs the system produced” [81] yielded “How fair and controllable is the system?” in our question set. We continuously moved back and forth between the texts and our preliminary categories when integrating new user questions to test our categories against the text passages.

As the outcome of this coding step, we (1) refined the categories elicited in the open coding, and (2) discovered differences and commonalities between the system qualities by grouping user questions according to the quality in focus of a paper (see Table 1 for an excerpt; please note that one question could be relevant for multiple system qualities).

4.2.3 Selective Coding: Building the Framework. In the last step, we identified core concepts overarching our categories. Practically, we wrote our user questions on cards we re-arranged on a large wall, plus example papers and stickers for possible core concepts. This set-up was then discussed extensively over several weeks and extended through further paper collection and reviewing. As a result, we identified and defined our framework dimensions presented in Section 5.

4.3 Limitations

Our framework should be understood and applied with several limitations in mind:

First, as work on interactive intelligent systems has evolved into a rapidly developing field in the last decade, future approaches might uncover solutions and principles that the presented framework does not cover in its current state. Second, this work focuses on eight system qualities, listed in Section 4.1. Since this set was shaped through our sampling process, we are confident that it covers a representative part of the diverse approaches and reflections in the field. However, it is not comprehensive. For a detailed overview of possible system qualities, see Abdul et al. [1]. Third, our paper set is mainly based on work accessible through the ACM Digital Library. We also included work published through IEEE, Springer, and Elsevier, among others, but only through snowball searches, not as a main database, to keep our paper set at a manually manageable size. Overall, these limitations might create the need to build on, extend and adapt the framework in future work.

Finally, this work focuses on HCI researchers as a target audience. In Section 6.3.1, we suggest ideas on how our framework might be applied to commercial systems and thus serve practitioners, too. However, these ideas still need to
be validated in the future, for example, through interviews with designers and developers, or by prototyping applications using the framework dimensions and evaluating them with end-users.

5 A USER-CENTERED FRAMEWORK FOR SUPPORTING USERS IN UNDERSTANDING INTELLIGENT SYSTEMS

Our analysis revealed three categories that capture and differentiate current assumptions about users and interaction with intelligent systems. They thus serve as a conceptual framework for structuring the discussion about supporting users in understanding intelligent systems from an HCI perspective, as presented in the next sections.

5.1 What do Users Seek to Know? User Mindsets

The user questions addressed in work on different system qualities imply different assumptions about what users seek to know: For example, searching for information about why a certain system output came into being [71] implies a very different kind of interest than wanting to know how well a program knows a given domain [84] or how a system was developed and continually improved [81]. Likewise, this is true for users asking how to correct system errors [65] compared to users that want to be merely informed about the presence of algorithmic decision-making [81].

To capture these differences, we introduce the category user mindsets. In psychological research, mindsets describe the “cognitive orientation” of people that precede the formation of intentions and planning of successive actions towards reaching a goal [38]. In the same manner, for this work we define user mindsets as users’ cognitive orientation that guides concrete intentions to interact with an intelligent system. From our analysis emerged three such mindsets that we find in prior work on the system qualities: utilitarian, interpretive, and critical, described in detail next.

5.1.1 Utilitarian Mindset. A utilitarian mindset aims to predict and control system behaviour to reach a practical goal. This mindset carries a strong notion of utility and/or usability.

Consequently, a utilitarian mindset is reflected by many examples in work on system qualities that imply a very practical view on user inquiry, such as in work on explainability and intelligibility. For example, users might want to understand system recommendations to better compare and find products they are interested in (Why was this recommended to me?) [80]. Moreover, users might want to train more effectively with an intelligent fitness coach [33], or understand a system ranking they financially depend on, as observed in AirBnB [51] or social media [18]. In another example, users worked more efficiently with a system feedforward based on What if? questions [22]. Similarly, user questions in work on scrutability such as How can I efficiently improve recommendations? [11] imply a utilitarian mindset.

On a meta-level, this mindset can also be found in work on interactive machine learning and end-user debugging. Research in these areas addresses user questions such as How can I assess the state of the learned concept? Where does the model fail? [31], How sure is the system that a given output is correct? [84], How to combine models? [91], Which model works best? [7], or How do changes affect the rest of the system? [66]. These and similar questions imply a focus on recognising and handling system error, giving feedback, or analysing the system to better work with it in the future.

5.1.2 Interpretive Mindset. An interpretive mindset strives to interpret system actions based on one’s perception of and experience with the system and its output. This mindset embraces the notion of user experience.

When users adopt this mindset, they do not necessarily want to reach a particular practical goal, but rather to understand the system based on a certain experience they have made. For example, a social media user might want to understand why posts of particular friends are not shown [18]. Moreover, an interpretive mindset might be adopted by
users who do not understand how they are being profiled, when they believe their feedback is not being considered or feel they lack control over system output [18].

Examples for an (implied) interpretive mindset can be found in work on transparency (e.g. How sensible – and not arbitrary or random – is the system? [81]) and interpretability (e.g. What else can the model tell me about the world? [72]). Moreover, it is reflected in many user questions articulated in work on scrutability, for example What else does the system think I (don’t) know?, What would the system do if I did Y?, or What does the system do for other people? [55]. Although control also plays an important role in research in this field, the underlying perspective is framed in terms of experience with and perception of the system and its output rather than a practical goal.

5.1.3 Critical mindset. A critical mindset stresses normative, ethical and legal reflection about intelligent systems.

This echoes the wider discussion about the system qualities, such as transparency, explainability and accountability (e.g. [34, 44, 74, 98, 99]). For example, a user might critique a system’s missing social intelligence [17, 18] or might want to know why it was developed in a certain way [81]. A critical mindset may thus be decoupled from system use.

Calls for support of critical inquiry are mainly found in work on system accountability. User questions include How was the system developed and how is it continually being improved?, What is informed by the intelligent system (i.e. reveal existence of intelligent decision-making and processing)?, How fair and controllable is the system? [81], How fair is a system decision? [13] or Can I trust this model? [83] and Should I trust this prediction? [42, 83].

User mindsets help to understand what users seek to know when interacting with an intelligent system. We should thus make explicit which mindset(s) we assume as a basis for our work (e.g. utilitarian, interpretive or critical).

5.2 How do Users Gain Knowledge? User Involvement

As introduced in the Scope and Foundations section of this paper, we assume that user understanding is built through the transmission of information between user and system.

Our analysis revealed that the great majority of work on the system qualities envisions this transmission of information in the form of a dialogue, that is as a “cycle of communication acts channelled through input/output from the machine perspective, or perception/action from the human perspective” [47]. Dialogue as an interaction concept inherently stresses the need for users to understand the system (and vice versa) [47]. It has even been argued that the characteristics of intelligent systems necessarily involve some sort of dialogue in order for users to understand them [84]. Elements of dialogue, such as structuring interaction as stages [47], are commonly found in work on the system qualities. Most notably, end-user debugging [61] and interactive machine learning [54] make use of mixed-initiative interfaces [48]. It thus seems that the system qualities almost imply this concept of interaction, so closely are they interwoven with a dialogue structure. To support users in understanding intelligent systems, information may thus be transferred in two directions, either from user to system, or from system to user. From a user perspective, this determines how users gain knowledge – either through action (active) or perception (passive).

Our second framework category, user involvement, captures these two ways of gaining knowledge. User involvement thus describes interaction possibilities to transfer information to or receive information from the system as a basis for user understanding. In the following sections, we distinguish work on the system qualities according to these two directions. In general, a system might support involvement in multiple ways (e.g. via explanation, controls, visualisations) and thus imply transitioning between both directions during use, for example, through interactive visual explanations [61].
5.2.1 Active User Involvement (User-to-System). User questions such as How can I tell the system what I want? [55], How can I detect system errors? [65], or What do I have to change to correct the system? [65] point to active users whose corrections and feedback are utilised by the system. The dialogue between system and user may be initiated by both sides and is then indeed based on turn-taking, as described earlier. For example, Alkan et al. [2] presented a career goal recommender that literally employs a dialogue structure to suggest items and incorporate user feedback. Work on scrutability by Kay and Kummerfeld [55] emphasises the “real effort” users must make when inquiring into a system. Other work on the system qualities, in particular in end-user-debugging and interactive machine learning, sees users in active roles as debuggers [63] or teachers [6].

Systems support an active user by offering interface controls tied to aspects of their “intelligence” (e.g. data processing, user model). These controls may enable users to experiment with the intelligent system (e.g. user model controls [55]). For example, if a user interface offers switches for certain data sources (e.g. in a settings view), users can actively experiment with the way that these data sources influence system output (e.g. switch off GPS to see how recommendations change in a city guide app; also see [55]). Moreover, Coppers et al. [22] introduced widgets for system feedforward that allow for active inquiry to answer What if? user questions such as What will happen if I click this checkbox? Another example is a separate, dedicated GUI for such experimentation, which allows users to directly set the values of certain system inputs and check the resulting output (e.g. “intelligibility testing” [71]).

Moreover, many visual explanations offer direct manipulation that also puts users into an active role: For instance, work in interactive machine learning proposed interactions with classifier confusion matrices to express desired changes in resulting decisions [54, 91]. Similarly, work on explanations for spam filtering enabled users to influence the classifier via interactive bar charts of word importance [61].

5.2.2 Passive User Involvement (System-to-User). User questions such as Why does the system think that I want/need X? [12], Why did the system do X? [69, 81], Why did it not do Y? [71] or How does the system produce an output? [81] suggest that users want to get informed about the systems inner workings, but do not actively provide the system with feedback and corrections. Users may still initiate the dialogue with the system, but are then restricted to be recipients of information. This way of user involvement is typically assumed by work on transparency and explainability, where displaying information about a system’s inner workings is a common tool for user support. For example, related work proposed visual and textual explanations that show how recommendations are influenced by data from customers with similar preferences [35]. Further examples of supporting user understanding in a passive way include icons that indicate “intelligent” data processing [33], interaction history [49], annotations for specific recommendations [14], and plots and image highlighting for classification decisions [83] or recommendations [93].

User involvement describes how user knowledge is built during interaction with a system. This depends on the direction of information transmission between user to system (e.g. users are involved in an active or passive way). We should explicitly state the nature of user involvement and how it is manifested in and supported through design.

5.3 Which Knowledge Do Users Gain? Knowledge Outcomes

The envisioned result of the different system qualities is knowledge that users gain about an intelligent system. However, this knowledge may refer to different aspects of the system and interaction, such as a specific recommendation [80] or the “reasoning” of a system [83]. To account for this variety of which knowledge users gain, we introduce our third framework category, knowledge outcomes. These characterise the nature of user understanding developed about an
intelligent system. Overall, our analysis surfaced four different knowledge outcomes currently addressed in the literature (output, process, interaction, and meta). Since output and process knowledge are often confronted in the literature, we present them in one subsection here, too.

These knowledge outcomes are not unique to HCI or intelligent systems. For example, output and process knowledge can be found in work on theory on gaining knowledge in practice [73]. Moreover, work on complex problem solving articulates output, process and structural knowledge [87], the latter being similar to our interaction knowledge.

We also introduce two qualities of knowledge emerging from the reviewed literature. Borrowing established terms for knowledge qualities in applied research theory [43, 73], we summarise them as rigour and relevance of knowledge.

5.3.1 Output and Process Knowledge. Output knowledge targets individual instances of an intelligent system (e.g. understanding a specific movie recommendation). In contrast, process knowledge targets the system’s underlying model and reasoning steps (e.g. the workings of a neural network that processes movie watching behaviour).

Explainability research in particular distinguishes between explanations for instances and models. For example, Ribeiro et al. [83] explain classifiers with regard to two questions, Should I trust this prediction? and Can I trust this model?. Therefore, they design for both output and process knowledge. These two knowledge types also motivate the what and how questions posed by Lim and Dey in their work on intelligibility [69] (e.g. What did the system do?). Also, Rana and Bridge [82] introduced chained explanations (called “Recommendation-by-Explanation”) to explain a specific output to users. Moreover, work on accountability makes system reasoning accessible to users to support the development of process knowledge [13].

5.3.2 Interaction Knowledge. Our knowledge type interaction knowledge describes knowing how to do something in an interactive intelligent system. For example, supporting users in gaining this type of knowledge motivates questions in work on scrutability (e.g. How can I tell the system what I want to know (or not)? [55]), interactive machine learning (e.g. How to experiment with model inputs? [6]), and end-user debugging (e.g. How can I tell the system why it was wrong? [65], How can I correct system errors? [63]).

5.3.3 Meta Knowledge. Meta knowledge captures system-related knowledge beyond interaction situations, such as information from a developer blog. For example, meta knowledge motivates some questions in work on transparency, such as How is the system developed and how is it continually improved? by Rader et al. [81]. They also explicitly add Objective explanations that inform users about how “a system comes into being” that result in meta knowledge (e.g. development practices and contexts). Moreover, this knowledge type is a main driver of work on accountability, in which computer science overlaps with journalism: For instance, Diakopoulus “seeks to articulate the power structures, biases, and influences” of intelligent systems [25].

5.3.4 Rigour and Relevance of Knowledge. Rigour: Kulesza et al. propose the concepts of soundness and completeness in their work on explanations in intelligent systems [61, 64] – soundness is truthful explanation, and completeness means explaining the whole system. Gilpin et al. [36] also refer to completeness, yet understand it as supporting anticipation of system behaviour in more situations. For an overarching view, we generalise this to a broader level: We regard soundness and completeness as facets of rigour. Linked back to the work by Kulesza et al. [61, 64], this means that a rigorous explanation, and the resulting understanding of a system, should be sound and complete.

Relevance: A rigorous understanding does not need to be useful. We argue that this aspect should be of explicit interest for a pragmatic HCI perspective. We thus consider relevance as another general quality of knowledge [73] that is crucial to make explicit in the specific context of user understanding of intelligent systems. This quality highlights our
pragmatic view: Elements like explanations are valuable if they add utility, that is, if they help users to gain knowledge that is relevant for using the system in better ways and towards better outcomes. In this pragmatic sense, this quality echoes Kulesza et al.’s suggestion to “not overwhelm” users with (irrelevant) information [61, 64]. What is relevant to know, and to which extent, may also depend on factors such as task and complexity of the system [19].

Knowledge outcomes and qualities characterise and make explicit what kind of user understanding a system seeks to facilitate. We should thus articulate the goals of our work (e.g. output, process, interaction, and meta knowledge) and reflect on rigour and relevance of the respective kind of knowledge.

6 FRAMEWORK APPLICATION: STRUCTURING PAST & FUTURE WORK

We have presented three categories for supporting user understanding of intelligent systems as emerged from our analysis of the literature – user mindsets, user involvement, and knowledge outcomes. These categories highlight differences and commonalities between work on the system qualities and serve as a conceptual framework of supporting users in understanding intelligent systems.

Our framework introduces an overarching user-centric structure to the field that abstracts from the fractured terminological landscape. We now propose to use our framework categories as a means for researchers in HCI and adjoining fields to clarify and make explicit the assumptions of their work, and to structure past and future work and discussions about how to support users in understanding intelligent systems. The boxes presented throughout this article provide inspiration on what to consider. The following sections highlight further applications of our framework.

6.1 Structuring Past Approaches and Solution Principles

In Figure 2, we demonstrate the application of our framework based on an interactive intelligent mood board creation tool by Koch et al. [58] as an example for system-aided design ideation. In their prototype, system and user collaborate to find suitable imagery, the system making suggestions which the user can accept or discard. For each suggestion, the system offers a textual explanation and different feedback options. In terms of user involvement, this approach thus supports both an active user (via options for feedback and correction and turn-taking), as well as a passive one (via information about why an image was suggested). With regard to knowledge outcomes, users gain relevant knowledge about a specific output (a specific image suggestion) and interaction knowledge about how to control future system suggestions (by telling the system what they like or not). Overall, the prototype is designed so as to best support users in the creation of a mood board as an ideation activity, and thus implies a utilitarian mindset.

Moreover, Table 2 presents other exemplary solution principles from the literature to illustrate how user mindsets, user involvement and knowledge outcomes may be used to structure past work. We do not claim to provide a comprehensive survey in our work, but selected these examples to show the diversity of approaches in our paper set.

Studying the approaches as arranged in Table 2 reveals interesting structures: For example, explanations appear across the charted space and thus could be seen as the go-to building block for many solution principles. They commonly provide output and process knowledge via text and/or plots [35, 49, 57, 61, 65, 70, 83]. Sometimes these representations also allow for interactivity and user corrections [61, 65, 71], in particular when explanations are referred to in work on scrutability, end-user debugging, and interactive machine learning [24, 54, 55, 91]. Explanations commonly arise from utilitarian mindsets, yet they also appear in work with interpretive and critical questions [14, 26, 55, 81].
6.2 Reflecting on Your Own Approach

6.2.1 Reframing User Questions. User questions are a helpful way to uncover users’ information needs. Our framework can be used to re-frame such questions in related work, in particular by reconsidering the underlying mindsets to view questions from a novel angle: For example, a question such as *Why did the system do X?* is currently mostly tied to a context implying a utilitarian mindset [69]. However, this question could also reflect other mindsets and thus different underlying user motives for inquiry, depending on the envisioned context. Design solutions to this question could then foster utilitarian (e.g. explain feature influences), interpretive (e.g. explain in terms of a user’s daily life context), critical (e.g. explain system decision given a community’s norms), or all three mindsets.

6.2.2 Explicating Perspectives with Mixed Mindsets. Related, reflecting on the three mindsets presented here can help to discover structure in perspectives that mix multiple mindsets: For example, a recent set of 18 guidelines for interaction with AI [8] contained mostly utilitarian guidelines (e.g. “Support efficient dismissal”, “Provide global controls”), yet two stand out as following a critical mindset (“Match relevant social norms”, “Mitigate social biases”). Our lens allows to clarify and explicate this mix, revealing, in this example, that the guidelines already follow a broader perspective than the related work itself alluded to with its stated focus “on AI design guidelines that [...] could be easily evaluated by inspection of a system’s interface”. Such analysis could help to structure discussions about similarly mixed perspectives.

6.2.3 Explicitly Determining Relevance. What is relevant to know for users is considered differently across the system qualities. Highlighting relevance and rigour (see section on *knowledge outcomes* and Table 1 last column) thus helps to reflect on what we consider important, for whom, and to what extent – and how we choose to determine it.

For example, explanation design often involves non-expert users, possibly via a user-centred design process [33] or scenario-based elicitation [71] to inform what is relevant, what should be explained, and to what extent.

In contrast, interactive machine learning focuses on information that is relevant to the task of the system (e.g. to improve a classifier), which is often operated by experts [6]. Therefore, what is relevant here (and to what extent) is
How to Support Users in Understanding Intelligent Systems? Structuring the Discussion

IUI '21, April 14–17, 2021, College Station, TX, USA

<table>
<thead>
<tr>
<th>Knowledge Outcomes</th>
<th>Process</th>
<th>Interaction</th>
<th>Meta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Users build if-rules in an editor, system detects problems via simulation (e.g. loops), user corrects them [94]. Natural language dialogue enables users to ask system what they could do next [92]. Users manipulate confusion matrix to change model [34], including re-combining multiple models [91].</td>
<td>Beyond runtime: Open source enables code audits and facilitates understanding of system [29]. Beyond single systems: Educate public in computational skills to facilitate system understanding overall [20].</td>
<td></td>
</tr>
<tr>
<td>Passive</td>
<td>Bu charts and image regions show importance of predictors to explain specific classification [85]. Visualisations show similar users’ input to explain recommendation [48]. Tree of images of training instances explains classification [100].</td>
<td>Beyond code: Educate/sensitise developers and decision makers to consequences of systems [20]. Beyond system use: People discuss and reflect on social implications and context of the system’s output [17, 86, 96].</td>
<td></td>
</tr>
<tr>
<td>Active</td>
<td>Rule-based reasoning system verbalises system decisions in natural language dialogue with user [36]. Separate profile page displays current user model [35]. Natural language dialogue enables users to ask system why it has not decided differently. [92] Constructivist learning: user manipulates system and updates mental model of it based on resulting changes in output [85].</td>
<td>&quot;Algorithmic E&quot; beyond interaction: Users engage in communication and relationship building with intelligent agents [78]. &quot;Algorithmic UX&quot; beyond system use: People learn from past interactions and output how to challenge system intelligence and its normative implications through unexpected or malicious input (e.g. manipulating public chatbot via Twitter) [75].</td>
<td></td>
</tr>
<tr>
<td>Passive</td>
<td>&quot;Algorithmic accountability reporting&quot;. Journalists report on black box systems, e.g. by trying out inputs systematically [25, 27]. Output shown with textual explanation of the decision process [57]. Animation indicates how system output is generated (e.g. dice roll for randomness) [55].</td>
<td>Beyond system understanding: People learn from past interactions and output how to challenge system intelligence and its normative implications through unexpected or malicious input (e.g. manipulating public chatbot via Twitter) [75]. Beyond system use: People discuss and reflect on social implications and context of the system’s output [17, 86, 96].</td>
<td></td>
</tr>
<tr>
<td>Active</td>
<td>Record models, algorithms, data, decisions for later audit [10]. Annotate recommended content pieces with indicators for quality/reliability of their source (e.g. for news) [67]. Textual explanations of system intelligence on a high level, not integrated into the system (e.g. articles about system) [41]. Explaining the logic behind an algorithm with another algorithm [16].</td>
<td>&quot;Algorithmic Imaginary&quot;: People develop understanding of intelligent systems and how to influence them based on how “they are being articulated, experienced and contested in the public domain” [18]. Textual explanations of developers’ intentions on a higher level, not integrated into the system (e.g. articles about system) [81].</td>
<td></td>
</tr>
<tr>
<td>Passive</td>
<td>&quot;Algorithmic profiling management&quot;: Profile page reveals what system knows and how this influences content, including past interactions, controls enable modifications [3]. Separate profile page enables users to edit what the system knows about them [15]. &quot;Algorithmic profiling management&quot;: Profile page reveals what system knows and how this influences content, including past interactions, controls enable modifications [3].</td>
<td>Icon indicates which system output is influenced by AI [39]. Text analysis extracts relevant sentences from reviews to show along with recommendation [28].</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Examples of approaches and solution principles for supporting user understanding of intelligent systems, structured through our framework. This is not a comprehensive survey; examples were selected to illustrate the diversity of approaches in the literature.

foremost informed by the machine learning task, and less so by studying or asking end-users. This can be seen, for example, in the UI elements derived in a recent survey on interactive machine learning [31], which are closely coupled to the machine learning task (e.g. they serve “sample review”, “feedback assignment”, etc.).

As another example, work on end-user debugging presents an action-focused middle-ground, between user-focused (as explainability) and system-focused (as interactive machine learning): Here, resulting knowledge should help users to make the system more relevant to them, for example, by correcting system errors from the users’ point of view; users may be both experts [7] or non-experts [65].
6.2.4 Informing Methodology. Our framework may be used to motivate methodological choices, for example when informing or evaluating the design of a new approach for supporting user understanding:

For instance, work catering to a utilitarian mindset might benefit from a controlled environment and precise measurements in a lab study. Even simulation of system decisions might be a (first) option [29]. A lab study might also be a suitable choice to evaluate support for developing interaction knowledge, since users can be directly observed during interaction (e.g. see [55, 91]). In contrast, if a design or research question targets an interpretive mindset and/or meta knowledge, it might be worthwhile or required to study user and system in their daily contexts of use (e.g. see lab vs field study in education context in [55]). The same holds for work motivated by a critical mindset, yet other methods exist here as well, such as online surveys or data analyses of views expressed through mass media and social network discussions [67], or policy and legal texts [10, 39, 79].

6.3 Going Beyond

6.3.1 Inspiring New Approaches in Research and Practice. Our framework may provide inspiration for new approaches and solution principles. While the target audience of this paper are foremost researchers in the field, we believe that the framework might also be used by practitioners, for example, as a brainstorming tool for prototype development.

We illustrate this using the proposed mindsets: UIs could support users in examining the system with different mindsets via “modes” for explanation views. Users could then switch between utilitarian explanations (e.g. explain a recommendation with product features) and interpretive or critical ones (e.g. explain system beliefs about a user, such as that the user is part of a certain target group; reveal that recommendations are assembled by an AI and not by a human, cf. [33]).

A more radical solution could offer three different views of the system that display or hide UI elements depending on the respective “mode”. Or the mindsets might simply help to decide which user approach to support in a system, and to identify those remaining unaddressed so far. Prototypes could then be tested with the respective end-users of the application. Yet, the actual generative power of our framework has to be validated in the future.

6.3.2 Understanding Systems Beyond Interaction Situations. The critical mindset and meta knowledge capture a crucial difference between traditional (non-intelligent) systems and what we see today and what is yet to come: Systems are increasingly interwoven with our lives, be it in everyday applications or in areas of consequential decision-making (e.g. financial, medical or legal). Their effects thus do not remain limited to a particular interaction situation. It is important that we as researchers reflect on the impact of the systems we design beyond the duration of direct use. This also includes reflections on when and how intelligent systems can learn compared to humans in the same roles [3]. Examples for work in such a larger context are presented in Table 2, in particular in the critical and meta areas (e.g. [9, 20, 25, 27]).

Connections with HCI in such work commonly refer to accountability and transparency of intelligent systems.

6.3.3 Motivating Connections Beyond HCI & Machine Learning/AI. We see recent calls for more joint research at the intersection of HCI and AI to improve system understanding [1]. However, this is mostly motivated by utilitarian or interpretive mindsets. Thus, another related key takeaway is to draw attention to interdisciplinary connections via the critical of the three mindsets proposed in this article: As is evident from recent “AI and data scandals” (e.g. [40, 67, 75]), developing more understandable (and accountable) intelligent systems also needs to be addressed in a wider view (cf. third wave HCI [15]), for example across networks of human and AI actors [9]. More generally, fruitful connections could span considerations from fields like journalism [25, 27] and communication [67, 75], policy [3], sociology [9] and education [20], and ethical and legal concerns [16, 34].
7 CONCLUSION

Intelligent systems tend to violate UI principles, such as predictable output [6, 31], which makes them difficult to understand and use. To address this, researchers, practitioners, policy-makers and the general public call for system qualities such as transparency, scrutability, explainability, interpretability, interactivity, and so on. However, these terms are often blurred and employed with varying interpretations. This impedes conceptual clarity of the very properties that are envisioned to foster users’ understanding of intelligent systems.

This review responds to this lack of conceptual clarity with an analysis and discussion of theoretical concepts and prototype solutions from the literature: We make explicit the diversity of different implied views on user mindsets, user involvement, and knowledge outcomes.

In conclusion, we provide researchers with a framework to (1) clearly motivate and frame their work, (2) draw connections across work on different system qualities and related design solutions, and (3) articulate explicitly their underlying assumptions and goals. With our work, we thus hope to facilitate, structure and advance further discussions and research on supporting users’ understanding of intelligent systems.

ACKNOWLEDGMENTS

This project is funded by the Bavarian State Ministry of Science and the Arts and coordinated by the Bavarian Research Institute for Digital Transformation (bid).

REFERENCES


