

# On Explanation

Thomas R. Roth-Berghofer, Michael M. Richter

The term “explanation” has been widely investigated in different disciplines such as cognitive science, knowledge-based systems, linguistics, philosophy of science, artificial intelligence, and teaching. All these disciplines consider certain aspects of the term and make clear that there is not only one such concept but a variety of concepts. This has as a consequence that there was no common agreement on how this term should be used. One of the reasons is that explanation is some kind of an umbrella term, which covers rather different kinds of explanations. As a consequence, the semantics of this term is ambiguous. In the following, we introduce the main participants in any explanation scenario and highlight some of the results of explanation research from the literature before we give an overview of the articles in this journal issue.

## 1 Explanation participants

As general explanation scenario we consider the following with three participants (cf. Figure 1):

- The system or an agent that provides something to be explained, e.g., the solution to some problem, a technical device, a plan, or a decision. We call this agent the *originator*. This agent is interested in which way the user reacts after receiving the explanation.
- The *user* who is the addressee of the explanation.
- The *explainer* who presents the explanation to the user. This agent is interested in transferring the intention of the originator to the user as correct as possible. The explainer chooses the form of the explanation and is responsible for the computational aspects as well as for organising a dialog if needed.

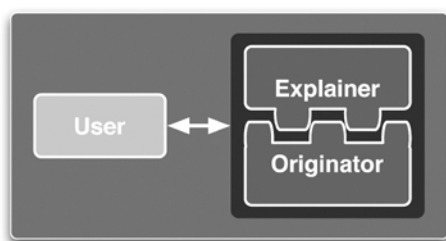


Figure 1: Participants in explanation scenario

The user applies the explanation in some way. With this application, in principle, a utility is connected, either for the user or for the originator. It depends on the application which utility dominates. Sometimes the user is primarily interested in reacting properly and sometimes the primary interest is on the side of the originator. The success of the explainer is measured in terms of these utilities. In any case, the utilities of both agents have to be modelled. In the past, mainly only user modelling took place. In a slightly more extended situation there are several agents in a discussion and giving explanations to each other. Here the roles of the participants are changing.

## 2 Explanations in the Literature

Explanations are in some sense always answers to questions, may they be raised or not. Therefore a specific topic of importance for our use of explanation is the logic of question and answer. The major point is now that an answer is not only simply true or false but it is rather important that it is an answer to the question. If there are only finitely many questions and answers then standard techniques suffice like frequently asked questions (FAQs) or forms that a customer can fill when, e.g., ordering a book. There a classification of question types and corresponding answer types has been developed (see [1] as a basic reference). If questions and answers are iterated then we call this a dialogue with explanatory character (see, e.g., [4]).

Major research on explanation components started in the early 1980ies and was strongly connected with the development of knowledge-based systems, called expert systems at that time. Many of the applications we have in mind are explanations of software systems. A generally recommended reference is [14].

An explanation component played already an important role in MYCIN [2], one of the first important expert systems and contributed significantly to the popularity of the system. The user was allowed to ask questions like “how was this achieved?” or “why have I to input these data?“. One of the major deficits in MYCIN was the inability to identify causal relations. Explanations in MYCIN were essentially intelligent and user friendly formulated / formatted parsing results. It can also be regarded as a task-based explanation component that is, however, not a generic task explanation system. Its tasks are at a much lower level than those associated with generic tasks.

XPLAIN [10] is a software product that helps users to build expert systems containing explanation components. It makes also knowledge about the domain usable. For this it allows the domain model to contain the facts of the domain as an input. The second form of input is a collection of domain principles, which are the methods or algorithms that apply to the facts. This system refines the domain knowledge (preserving the knowledge) until it is at an appropriate level for the implementation of an expert system. An extension is ESS (Explainable Expert System, see [11]). Here knowledge about concepts and concept classes can be formulated.