

A Taxonomy for Generating Explanations in Recommender Systems

Gerhard Friedrich and Markus Zanker

Alpen-Adria-Universitaet Klagenfurt
Universitaetsstrasse 65-67
9020 Klagenfurt, Austria

Abstract

In recommender systems, explanations serve as an additional type of information that can help users to better understand the system's output and promote objectives such as trust, confidence in decision making or utility. This article proposes a taxonomy to categorize and review the research in the area of explanations. It provides a unified view on the different recommendation paradigms, allowing similarities and differences to be clearly identified. Finally, the authors present their view on open research issues and opportunities for future work on this topic.

Introduction

Typical explanations for product recommendations include constructions such as “the digital camera *Profishot* is well-suited to your needs because you would like to take pictures of your children playing soccer” or “it is a light-weight compact camera especially designed for action photos”, or in the movie domain “the film *Beyond-SF* was extremely well received by science fiction fans and so you will probably enjoy it too”. Such information is commonly exchanged between a sales assistant and a customer during in-store recommendation processes and is usually termed an explanation (Brewer, Chinn, and Samarapungavan 1998).

We define explanations in recommender systems (RSs) by two properties. First, they are information about recommendations. A recommendation is typically a ranked list of items. Second, explanations support objectives defined by the RS designer. (Tintarev and Masthoff 2011) provide an enumeration of potential objectives that can be intended by a recommender system's explanations.

For example, the intention behind disclosing the reasoning process of the system could be to increase the user's confidence in making the right decision or to provide additional information such that the user can validate the rationality of the proposed purchase. Indeed, explanations can differ significantly, for instance, they may attempt to maximize the user's confidence throughout the shopping experience while ignoring satisfaction after purchase. Thus, the possible objectives of explanations are manifold, including aims such as

increasing trustworthiness, effectiveness or persuasiveness just to name a few.

This article's contribution is a taxonomy that structures the abundance of work that has produced in the area. We will therefore discuss explanations from different viewpoints and provide an overview of opportunities for future research.

Taxonomy for generating explanations

In the following section we will categorize different approaches for explaining recommendations based on major design principles. Figure 1 sketches the three dimensions of the taxonomy, namely the *reasoning model*, the *recommendation paradigm* and the *exploited information categories*, graphically.

Note, these three dimensions do not exhaustively capture all of the factors that determine the generation of explanations for RSs. For instance, the argumentation traces of explanations can be analyzed and classified as has been researched in the domain of expert systems by (Ye and Johnson 1995). However, complex argumentation chains that support each claim by justifying and backing statements are typically based on non-monotonic logical formalisms (Chesñevar, Maguitman, and Loui 2000) whose applicability in highly interactive online environments requires further investigation. In traditional decision support systems (DSSs) (Gönül, Önkal-Atay, and Lawrence 2006) the *presentation format* and the *provision mechanism* are also used to classify explanations. Text, images, video and combinations thereof can be used to explain a system's output. The provision mechanism describes who initiates the generation of explanations. In RSs explanations can be either explicitly requested by users (i.e. *user-invoked*) or automatically displayed by the system. Furthermore, structural characteristics such as *length*, *writing style* (for instance, the system could use flattering or more factual phrases) or the *confidence* that is conveyed in explanations can be used as additional dimensions. However, we restrict our taxonomy for generating explanations to those dimensions that are already in use in the literature of the field.

In contrast to classification by design principles, explanations can be categorized by their effects or impact on their users (Tintarev and Masthoff 2011). The impact of explanations on users is measured in terms of the achievement of different objectives that are therefore endogenous constructs

(in contrast to the aforementioned exogenous design principles). However, the influence of different explanation strategies with respect to the achievement of objectives such as user satisfaction or trust has not yet been systematically researched. However, one can classify explanation strategies by their intended goals.

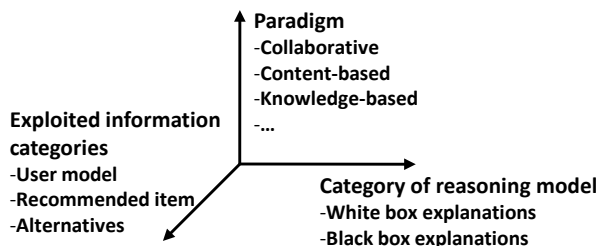


Figure 1: Dimensions for categorizing explanation approaches.

Categories of reasoning model for generating explanations

The reasoning model describes how explanations are generated and is the most fundamental distinctive criterion. Conventionally, the notion of “explaining recommendations” indicates that the system makes its reasoning process transparent to its user. Looking back in AI’s history, expert systems such as the well-known MYCIN (Buchanan and Shortliffe 1984) differentiated between *how* and *why-explanations*. In cases where the expert system requires input from the user, explanations as to why this information is needed are provided. Conversely, when an expert system outputs a solution, a client might ask for a justification of the proposed solution. Such explanations are called *how-explanations* because classical expert systems exploited information as to how a conclusion was deduced, for example, the sequence of rules activated in the decision-making process. The implicit assumption was that the model employed to give recommendations can also serve as a model for the argumentation that the advice is plausible from the client’s viewpoint.

However, such traces of deduction are not always accepted as high quality explanations, typically for two reasons. First, depending on the knowledge base, the deduction traces may be far too complex and may confuse the client. This motivated research into high level explanations (Squalli and Freuder 1996). Second, the rules exploited for deduction are not necessarily accepted as valid arguments for approving a solution. This argument is based on the observation that the actual production rules in various expert systems only shallowly reflected the known principles and laws of the underlying domain. E.g. experience based rules were formulated for linking symptoms with faults although physical laws exist which could be employed both for drawing conclusions and for justifying the solution based on first-principles. The essence of this discussion about deep versus shallow knowledge (Chandrasekara 1991) is that it is the content of knowledge bases and their empirical ground-

ing that are important and not syntactical representation like rules or constraints.

Analogously, similar observations can be made for recommender systems: although the various recommender system approaches agree on the task, the methods exploit different types of information and employ different deduction principles for generating recommendations, i.e. collaborative vs. content-based vs. knowledge-based just to mention the three archetypes. For instance, collaborative approaches exploit the similarities between users and items. Knowledge-based recommenders exploit known dependencies between properties of items and users. For instance, someone with a large family and income will prefer large houses. Therefore, an explanation approach which is based on the exploitation of the recommender’s reasoning model must be different for the above mentioned classes of recommendation systems. However, most of the recently proposed recommendation algorithms that are for instance based on matrix factorization models or ensemble learning are unable to explain their recommendations to their users. Herlocker et al. (Herlocker, Konstan, and Riedl 2000) therefore differentiate between white box and black box explanations. White box explanations disclose and exploit the underlying conceptual model of the recommendation engine, while black box explanations do not disclose the functioning of the system to the user. Reasons for the latter include that the process of computation should not be revealed or there is no sensible way to convey complex computations. For instance, Vig et al. (Vig, Sen, and Riedl 2009) use tags to explain that the recommended item and the user’s profile are semantically close while Zanker and Ninaus (Zanker and Ninaus 2010) dynamically combine canned text to generate knowledgeable explanations, although the recommendations have been identified and ranked with a different mechanism. To summarize, black box explanation approaches compute justifications that argue why a specific recommendations is plausible or should be of interest even though the reasoning model did not consider these propositions when actually computing the recommendation. In contrast, white box explanations disclose (at least partially) the reasoning model and its content to the user in the tradition of expert system’s *how-explanations*.

Recommendation paradigms

Collaborative and content-based filtering as well as knowledge-based recommendations constitute the three basic recommendation paradigms¹ (Jannach et al. 2010). Despite their differences in computing the recommendations, they share the same output function (i.e. a ranked list of items) and some ontological commitment which helps to unveil commonalities and generate a unifying view. All personalized recommender systems employ the concepts of *items* and *users*. These concepts, which are effectively sets of entities, are related by predicates (sometimes called associations or roles), such as *likes(Users, Items)* or

¹Note, other authors use slightly different grouping, such as collaborative, content-based and hybrid (Adomavicius and Tuzhilin 2005).

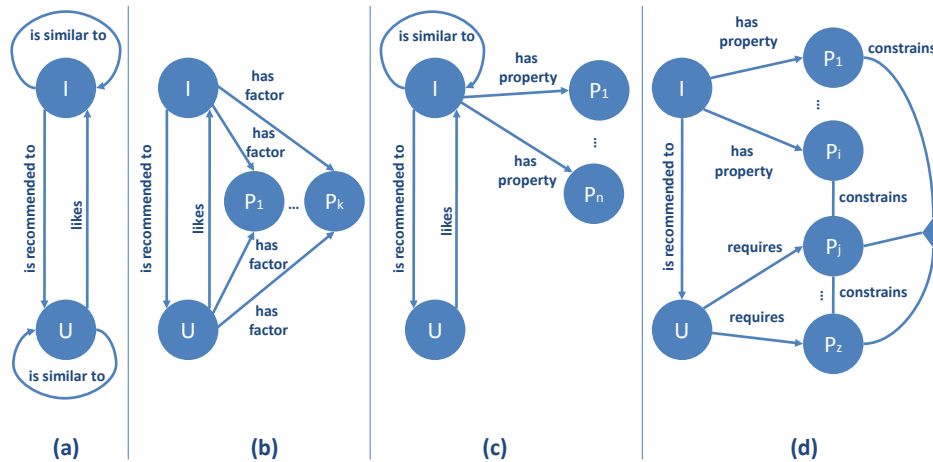


Figure 2: Archetypes of domain knowledge.

$recommendedTo(Items, Users)$. In addition, properties can be exploited to characterize concepts like items and users. These may also be categorized into subconcepts, e.g. *price*, *tags*, *degreeOfSatisfaction* or other sets of feature values. These properties are then employed in relations to describe a domain. Figure 2(a) illustrates the domain of collaborative filtering, where preference relations between users and items are known (e.g. encoded by a rating matrix). Neighbourhood-based collaborative filtering algorithms can extend such models by computing either similarity relationships between users or items. In the recommendation phase these similarity relationships are exploited to recommend items that are either similar to the items a user has liked in the past or that are liked by similar users. Specific variants of collaborative filtering, known for their very accurate rating predictions, are matrix factorization models (Koren, Bell, and Volinsky 2009). The basic model of matrix factorization (MF) identifies k abstract factors that can explain most of the signal in the ratings. Figure 2(b) depicts these factors as abstract property classes $P_1 \dots P_k$. Explanations of collaborative models disclose similarity relationships between concept instances. However, abstract properties, such as those identified by MF recommenders, rarely capture semantic meaning that can easily be communicated to users. Nevertheless, for the sake of transparency, a MF recommender might therefore compute similarity relationships between users and/or items based on its factor space representation.

The archetype of a content-based RS is depicted in Figure 2(c). There, knowledge about items is given, for instance in the form of a product catalog with item descriptions or by a term frequency vector. Recommendations are computed by determining those items that are most similar to items the user is already known to like. Content-based white-box explanations therefore explain similarities between items by disclosing their property relationships.

The third paradigm encompasses knowledge-based recommendation approaches. They are characterized by ad-

ditional domain properties such as abstract users requirements or preferences as well as various relationships between them. They, for instance, encode explicit sales expertise like which item features help to fulfill a specific user requirement. For instance, a constraint-based RS models items and users using sets of variables (represented by $P_1 \dots P_i$ and $P_j \dots P_z$ in Figure 2(d)). N-ary relationships between these properties constrain the allowed combinations. Solutions are computed by identifying those items that satisfy all the domain restrictions. In cases where no solution exists, the user is either asked to revise her/his preferences or the reasoner relaxes some of the domain restrictions. See (Zanker, Jessenitschnig, and Schmid 2010) for a discussion of different strategies for preference reasoning in constraint-based RS. Consequently, explanations disclose variable assignments (or property and requires relationships) and the constraining relationships between them to explain a recommendation (Friedrich 2004). Utility-based RSs that, for instance, implement Multi-Attribute-Utility Theory (MAUT) (Felfernig et al. 2008) also fall into this category.

Consequently, the unifying view is that explanation generation (like the computation of recommendations itself in case of a white box approach) exploits the relations between users, items and properties. However, how the instances of these relations are derived depends on the employed paradigm. For instance, the fact that two items are similar can be deduced using most of the aforementioned techniques, but the explanation for such a specific similarity relationship and the associated reasoning principles differ depending on the applied reasoning paradigm.

Information categories

The third characterizing dimension is comprised of the *information categories* that can be exploited for generating explanations. We differentiate between three different aspects of input:

1. *User model*. Are explanations tailored to the system's beliefs about the given user? For instance, the system could

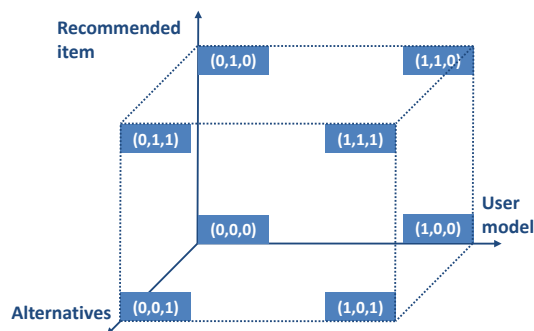


Figure 3: Information categories.

present arguments based on the user’s known ratings, preferences or demographics.

2. *Recommended item.* Is the explanation dependent on the specific recommended item? For instance, does the explanation make statements about the specific characteristics of the recommended item?
3. *Alternatives.* Do explanations argue in favor of or against alternatives to the recommended item?

Obviously, explanations can be tailored to all three aspects, none of them or any other combination as depicted in Figure 3. For instance, explanations located at position (0,0,0) do not adapt to any of the three information categories and can be realized using static text. E.g. explanatory phrases such as “the system made correct predictions in over 80% of all sessions” or “users with similar page views like you, purchased the following products”.

Examples that only address the user model in generated explanations (position (1,0,0)) include “we considered the rating profiles of the 48 most similar peers from Chicago” or “your average rating is 3.5 stars”. In contrast, explanations that refer only to known data on the recommended item could be as follows: “the proposed movie won 3 Golden Globe and 2 Oscar Awards” or “this wine is the perfect accompaniment for desserts”. However, most commonly one expects explanations to exploit both the user model and characteristics of the recommended item such as a histogram that depicts “your neighbours’ ratings for this movie” or “this digital camera is very reliable and is the perfect companion for going abroad, however it costs a bit more than you specified”. The third aspect enables explanations to compare a recommended item with alternative options and argue with their differences. For instance, Pu and Chen (Pu and Chen 2007) researched designs providing explanation such as “although this is the most popular product, we would also recommend the following products that are cheaper and lighter, but have lower processor speeds”.

Finally, an explanation that encompasses all three aspects together might include the following: “We think you will enjoy the Carribean Spa Resort as it offers, for instance, romantic candle light dinners that are perfect for young couples. Furthermore, this resort is closer to your home and less costly than the second ranked offer”.

Commonalities of explanations

An important principle of explanations is succinctness which has a long history in knowledge-based systems. Information and knowledge which is not relevant for answering a request should be excluded from an explanation. The historic MYCIN expert system also included this feature, only disclosing rules which actually contributed to an answer. In constraint-based systems only those constraints which are needed to generate the value assignments for entailing a recommendation are considered for explanations. In general, generating explanations can be seen as a form of abductive reasoning (Eiter, Gottlob, and Leone 1997), where the goal is to find the smallest subsets of a set of logical sentences such that a query is entailed. For example recommending a car to someone who has many children and does not care about driving performance will be based on the number of seats and the volume of the car. Therefore, arguing based on horsepower is irrelevant and should thus be excluded from an explanation. More formally, given a knowledge base KB and a query Q (a.k.a. recommended item in this context) that should be explained, the smallest subset KB' that entails Q is the basis for an explanation, i.e. $KB' \subseteq KB, KB' \models Q$ and for all $KB'' \subset KB', KB'' \not\models Q$ where \models is the usual entailment relation for logical sentences.

Note, several explanations can exist for a recommendation. In addition to succinctness, systems generating explanations must consider the presented solutions to avoid spurious explanations (Friedrich 2004). This minimal knowledge base serves as a basis for generating explanations, but is not one itself because, in the context of recommender systems, explanations are the inputs for building arguments for humans. Some parts of the knowledge base may be necessary for entailment but are obvious for humans. For instance, classifying a van with 7 seats as a family car is obvious for most users whereas a RS has to deduce this classification from data sheets in cases where this is not given.

Furthermore, this principle of succinctness and abductive reasoning can be applied to all forms of recommender paradigms. First, collaborative and content-based recommender systems are designed to exploit all relations of the employed ontology for their reasoning process. So the set of relations is the smallest required to derive a recommendation by design. However, the set of relation tuples needed to provide a specific recommendation is, in fact, reduced for the generation of explanations in various systems. For instance, in (Herlocker, Konstan, and Riedl 2000) explanations of the following type are considered: “As 23 out of 33 people similar to you like this item, we think you will like it too”². Note, the set of similar users is reduced to those relevant for the recommendation. In addition, the identities of individuals unknown to the user are not explicitly shown. Instead, aggregated values are presented because one could assume that a greater level of detail would be meaningless and confuse the user.

²This is a textual description of a graphical representation contained in (Herlocker, Konstan, and Riedl 2000)

Table 1: Categorization of explanation approaches

REFERENCE	EXPLANATION STYLE/ SYSTEM	REASONING MODEL	PARADIGM	INFORMATION CATEGORIES*
(Herlocker et al. 2000)	Histogram with grouping Past performance	White box White box	Collaborative Collaborative	(1,1,0) (1,0,0)
(Jannach 2004)	Advisor Suite	White box	Knowledge-based	(1,1,0)
(Bilgic and Mooney 2005)	Keyword style	White box	Content-based	(1,1,0)
	Neighbor style	White box	Collaborative	(1,1,0)
	Influence style	White box	Collaborative	(1,1,0)
(Pu and Chen 2007)	Tradeoff explanations	White box	Knowledge-based	(0,1,1)
(Symeonidis et al. 2008)	MoviExplain	White box	Content-based	(1,1,0)
(Felfernig et al. 2009)	Plausible repairs	White box	Knowledge-based	(1,0,0)
(Vig et al. 2009)	Tagsplanations	Black box	Content-based	(1,1,0)
(Zanker and Ninaus 2010)	Knowledgable expl.	Black box	Knowledge-based	(1,1,0)

* (User model, Recommended item, Alternatives)

Exemplary practical systems

Finally, we instantiate the proposed taxonomy, by presenting exemplary work in the field. Table 1 summarizes the categorization of different approaches. The paper of (Herlocker, Konstan, and Riedl 2000) is the first and most influential one (based on citation count) related to this topic. They compared 21 different explanation styles for collaborative filtering with a within participants experimental design and asked users about their intent to see the recommended movies. We denote the best and the second best strategy according to measured user intention in Table 1. Their paradigm is collaborative because data about the similarity relationships between users is disclosed. However, explanations may present this data with varying levels of detail by considering or ignoring data from the current user or the recommended item. (Bilgic and Mooney 2005) compared content-based and collaborative explanation styles, i.e. argumentation based on keywords describing the recommended item vs. disclosing the behavior of the most similar peers. Their evaluation measured how informative the explanations appeared to users by comparing user ratings for the recommended items after having read the explanations with re-ratings for the same items after having experienced them. Not astoundingly, the keyword style performed best in this setting. (Symeonidis, Nanopoulos, and Manolopoulos 2008) further developed the keyword and influence explanation styles by establishing a collaborative and content-based hybrid that justifies recommendations as follows: “Item A is suggested because it contains features i and j that are also included in items X, Y and Z that you like.” Consequently, by disclosing the similarities between items based on their properties the generated explanations follow a content-based argumentation approach that adapts to what is known about the current user and the recommended item. A somewhat similar but more encompassing idea has been proposed by (Vig, Sen, and Riedl 2009) who select the presumably most influential (user generated) tags instead of keywords to justify recommendations. Furthermore, they apply this explanation technique as a black box approach where state-of-the-art collaborative filtering mechanisms compute the recommendations themselves.

In the category of knowledge-based recommendation systems the Advisor Suite system (Jannach 2004) is an example of a system that discloses the sets of successfully applied and relaxed filter constraints to its users. Felfernig et al. extend such conversational recommendation systems by proposing a mechanism to compute repair actions that explain to users how to modify their specified requirements in order to avoid empty recommendation lists (Felfernig et al. 2009). In (Zanker and Ninaus 2010) the generation of explanations from declarative knowledge bases is generalized to a predicate-based finite state automata that is meant for the sole purpose of flexibly composing explanations from canned text.

Finally, tradeoff explanations provide additional information about alternatives by comparing the features of a recommended item to the characteristics of other items, i.e. addressing the information category *alternatives* in our taxonomy. (Pu and Chen 2007) developed a trust model and report on the potential of these explanation interfaces in building trust relationships between such a system and its users.

Open research issues

Currently most implementations of explanation components in recommender systems follow a white box approach. This is reasonable in order to optimize development effort. However, from a scientific point of view little research concentrates on predicting which explanation strategies are best suited for achieving which explanation objectives. Consequently, we anticipate substantial improvement in the quality of explanations, provided that researchers develop a better understanding of how different explanation traits impact users’ in order to develop an effective explanation model for particular domains.

Basically, we see two lines of research towards more effective explanations. First, the creation of new kinds of information, interaction and presentation styles which can be exploited for generating explanations. Second, analysis of how this information, interaction and presentation affect various explanation objectives and under which conditions. E.g. the development of a theory that classifies which types of users,

situations, items and explanation objectives favor a particular explanation model.

The development of various kinds of explanations is affected by the current technical state of the art. For instance, social networks and the increased provision of information by customers (e.g. tags, reviews and other forms of user generated content, geospatial data, friendship relations, linked data etc.) will drive future research towards novel data mash-ups and interaction scenarios. Thus, Web 2.0 and the Semantic Web will enable many new forms of explanations.

In terms of theory, we expect that insights from related disciplines like psychology and marketing will help to systematically explore the impact and tradeoffs of different aspects of explanation strategies. Psychology has elaborated a rich body of knowledge which can be exploited in recommender systems to achieve explanation objectives. For instance, framing and formulating decision problems in different ways produces predictable shifts of preference (Tversky and Kahneman 1981). As explanations (at least partially) help to construct a user's decision frame, they need to be carefully designed to consider these effects. In (Gönül, Önkal-Atay, and Lawrence 2006) a significant interaction effect between the length and the conveyed confidence in explanations of DSS is reported. One of the article's findings is that users appreciate long textual explanations only if the system conveys a strong confidence in its formulation. See (Teppan and Felfernig 2009; Yoo and Gretzel 2011) for further discussions on the application of different psychological effects in RSs leading to persuasive system traits.

Furthermore, if there is a clear understanding of the conditions under which a specific explanation form is most effective, it is reasonable to design methods that increase the likelihood of generating the right explanation by reducing the uncertainty about users' mental states. E.g. if we know that a certain type of customer favors specific types of arguments, methods to classify the customer are of increased importance.

Although it is desirable to develop a perfect theory that can predict the most appropriate explanations for all contexts and situations of online recommendations, we anticipate that developers of recommendation systems still have to accept some uncertainty with respect to the quality and impact of their designed explanation strategies. Consequently, the question is how observations from interactions with conversational recommender systems can be exploited to improve explanations and subsequently the overall performance of the recommender system itself. One initial piece of work in this direction is reported in (Mahmood, Ricci, and Venturini 2009). There different conversational moves of a RS (and different explanation strategies definitely constitute such moves) are interpreted as actions in a probabilistic state model and the system's performance can be optimized with the aim of reaching some preferred goal states.

To sum up, there is plenty of evidence that explanations are a powerful tool for influencing user behavior. With the increased importance of recommender systems in the sales process, interest in explanations is increasing. AI methods and Web 2.0 technologies combined with insights from neighboring fields have already developed several success-

ful explanation approaches. However, given that we have only exploited a small fraction of the available knowledge about human behavior, we anticipate many interesting and valuable research results related to this topic in the near future.

References

- Adomavicius, G., and Tuzhilin, A. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6):734–749.
- Bilgic, M., and Mooney, R. J. 2005. Explaining recommendations: Satisfaction vs. promotion. In *Workshop: Beyond Personalization, held in conjunction with the 2005 International Conference on Intelligent User Interfaces*, 13–18.
- Brewer, W. F.; Chinn, C. A.; and Samarapungavan, A. 1998. Explanation in scientists and children. *Minds and Machines* 8(1):119–136.
- Buchanan, B., and Shortliffe, E. 1984. *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley.
- Chandrasekara, B. 1991. Models versus rules, deep versus compiled content versus form: some distinctions in knowledge systems research. *IEEE Expert* 6(2):75–79.
- Chesñevar, C. I.; Maguitman, A. G.; and Loui, R. P. 2000. Logical models of argument. *ACM Comput. Surv.* 32(4):337–383.
- Eiter, T.; Gottlob, G.; and Leone, N. 1997. Abduction from logic programs: Semantics and complexity. *Theor. Comput. Sci.* 189(1-2):129–177.
- Felfernig, A.; Friedrich, G.; Teppan, E.; and Isak, K. 2008. Intelligent debugging and repair of utility constraint sets in knowledge-based recommender applications. In *Proceedings of the 13th ACM International Conference on Intelligent User Interfaces*, 218–226.
- Felfernig, A.; Friedrich, G.; Schubert, M.; Mandl, M.; Mairitsch, M.; and Teppan, E. 2009. Plausible repairs for inconsistent requirements. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, 791–796.
- Friedrich, G. 2004. Elimination of spurious explanations. In de Mántaras, R. L., and Saitta, L., eds., *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI)*, 813–817. Valencia, Spain: IOS Press.
- Gönül, M. S.; Önkal-Atay, D.; and Lawrence, M. 2006. The effects of structural characteristics of explanations on use of a dss. *Decision Support Systems* 42(3):1481–1493.
- Herlocker, J. L.; Konstan, J. A.; and Riedl, J. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work (CSCW'00)*, 241–250. Philadelphia, Pennsylvania, USA: ACM.
- Jannach, D.; Zanker, M.; Felfernig, A.; and Friedrich, G. 2010. *Recommender Systems An Introduction*. Cambridge University Press.

- Jannach, D. 2004. Advisor suite - a knowledge-based sales advisory system. In de Mantaras, R. L., and Saitta, L., eds., *Proceedings of European Conference on Artificial Intelligence*, 720–724. Valencia, Spain: IOS Press.
- Koren, Y.; Bell, R. M.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *IEEE Computer* 42(8):30–37.
- Mahmood, T.; Ricci, F.; and Venturini, A. 2009. Improving recommendation effectiveness: Adapting a dialogue strategy in online travel planning. *J. of IT & Tourism* 11(4):285–302.
- Pu, P., and Chen, L. 2007. Trust-inspiring explanation interfaces for recommender systems. *Knowl.-Based Syst.* 20(6):542–556.
- Ricci, F.; Rokach, L.; Shapira, B.; and Kantor, P. B., eds. 2011. *Recommender Systems Handbook*. Springer.
- Sqalli, M. H., and Freuder, E. C. 1996. Inference-based constraint satisfaction supports explanation. In *AAAI/IAAI, Vol. 1*, 318–325.
- Symeonidis, P.; Nanopoulos, A.; and Manolopoulos, Y. 2008. Providing justifications in recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part A* 38(6):1–1272.
- Teppan, E. C., and Felfernig, A. 2009. Asymmetric dominance- and compromise effects in the financial services domain. In Hofreiter, B., and Werthner, H., eds., *CEC*, 57–64. IEEE Computer Society.
- Tintarev, N., and Masthoff, J. 2011. Designing and evaluating explanations for recommender systems. In Ricci et al. (2011). 479–510.
- Tversky, A., and Kahneman, D. 1981. The framing of decisions and the Psychology of choice. *Science* 211:453–458.
- Vig, J.; Sen, S.; and Riedl, J. 2009. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent user interfaces, IUI '09*, 47–56. New York, NY, USA: ACM.
- Ye, L. R., and Johnson, P. E. 1995. The impact of explanation facilities in user acceptance of expert system advice. *MIS Quarterly* 19(2):157–172.
- Yoo, K. H., and Gretzel, U. 2011. Creating more credible and persuasive recommender systems: The influence of source characteristics on recommender system evaluations. In Ricci et al. (2011). 455–477.
- Zanker, M., and Ninaus, D. 2010. Knowledgeable explanations for recommender systems. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, 657–660.
- Zanker, M.; Jessenitschnig, M.; and Schmid, W. 2010. Preference reasoning with soft constraints in constraint-based recommender systems. *Constraints* 15(4):574–595.