An Overview of Explanations in Reasoning Systems

Literature Review

Solomon M. Malesa
mlssol001@myuct.ac.za
University of Cape Town
Computer Sciences (Honours)
South Africa

ABSTRACT

Reasoning Systems are systems that generate new conclusions from a Knowledge Base using some form of logic. Although there is an increasing trend of adopting Reasoning Systems as primary advisers for crucial decisions made in industries such as the medical industry, financial sector and government, the acceptance of these systems is hindered by the lack of explanation for actions taken by such systems. In this survey, we explore how explanation facilities in Reasoning Systems add value to the user's experience by explicating the decisions of Reasoning System. We know that an explanation is the set of premises that are required to be true in order for a conclusion to hold. Thus in this survey, we begin by defining what constitute a 'good' explanation and what types of explanations are available. We take a closer look into the factors that impact explanation designs. After having set the benchmarks for explanation facilities, we explore different applications of explanation facilities in Recommender Systems, Symbolic Reasoning Systems and Non Symbolic Reasoning Systems.

KEYWORDS

explanation, knowledge representation and reasoning, artificial intelligence

ACM Reference format:
Solomon M. Malesa. 2019. An Overview of Explanations in Reasoning Systems. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 6 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 BACKGROUND AND INTRODUCTION

A Knowledge Based System (KBS) is a system that uses a Knowledge Base (KB) consisting of propositions (which are interpreted to be true) to generate new knowledge that is consistent with the KB. The results of user queries generated by a KBS are not guaranteed to be accepted by the user since the steps taken to generate this "new" knowledge are usually hidden from the user. This lack of transparency is usually attributed to the fact that when KBSs are built, for example machine learning systems, the engineers are focused on the correctness and accuracy of the results and ignore explaining the decisions of the system to the user. According to Guidotti et al.[14] the inability to obtain an explanation for the decisions made by a KBS is an overwhelming drawback of the usability of KBSs.

As discussed by McGuinness and Patel-Schneide [17] conclusions that are not accompanied by an adequate explanation are puzzling to the users and are identified as a usability issue. Furthermore, Swartout and Moore [22] have identified the following factors that characterize the causes of limits in existing explanation facilities:

- **Narrow**: Existing KBSs allow a limited variety of queries.
- **Inflexible**: Existing KBSs are limited in their presentation of results (i.e. they can only return answers in one form). For example, some KBSs only return answers in a text format and not in a graphical format.
- **Insensitive**: Explanations of existing KBs cannot be tailored according to user needs.
- **Unresponsive**: Existing KBSs are not able to answer follow-up questions and cannot offer alternative explanations if the user is not satisfied with an explanation.
- **Inexstensible**: Users cannot extend or modify the explanation facilities of existing KBSs according to their needs.

To evaluate the usefulness of an explanation, we have to ensure that the quality of the explanation is such that it is acceptable to the user. This requires us to characterize what constitutes a 'good' explanation. Swartout and Moore [27] have identified the following five aspects of a good explanation in a KBS :

- **Fidelity**. Swartout and Moore [27] maintain that "The explanation must be an accurate representation of what the expert system really does".
- **Understandability**. It is obvious that for an explanation to be good, it must be understood by the user. Therefore, the user must be familiar with the terminology that is used by the KBS. The KBS must take into account the user's personal experience (e.g. their knowledge, concerns, goals and preferences). Not only should the KBS supply explanations at different levels of abstractions, but it should also be able to provide explanations with different levels of details and perspectives. The KBS must be able to provide explanations that are sound and "natural". Finally, the KBS must be able to further explain certain parts of its answers that users find to be confusing.
- **Sufficiency**. The KBS must contain enough information in order to be able to generate the required explanations.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

- **Low Construction Overhead**. The difficulty of constructing explanation facilities must be reasonable (in a sense that it should not be more difficult to construct an explanation facility than constructing the KBS).
- **Efficiency**. Swartout and Moore [27] claim that "The explanation facility should not degrade the runtime efficiency of the KBS".

Since not all explanations are of the same type, Swartout and Moore [27] have identified five different types of explanations that are common in the field of Knowledge Representation and Reasoning:

- **Explanations about the the systems behavior**. These are explanations which outline the reasoning behind the system's conclusions by paraphrasing the actions it took step by step[27].
- **Justifications**. Explanations of this kind provide the rationale behind the KBS conclusions[27].
- **Preferences**. Explanations of this kind provide the rationale behind why certain explanations are more valuable than others.
- **Domain explanations**. These define the problem domain itself[27]. An example of explanations of this kind are the explanations generated by the Digitalis Advisor, a program designed to give advise to physicians about therapeutic treatments and recommendations[28].
- **Terminology definitions**. These types of explanations are focused on the definitions of the terminology used by a KBS [27]. For expample, the system may have to answer a query such as "What is a compounded drug".

2 RESEARCH IN EXPLANATION DESIGN

The purpose of explanations is to supplement the user's understanding of conclusions made by a KBS. Darlington [11] recognises that this requires a focus on the two parties involved: A user of the system, the system itself (usually called an expert system). The relationship between the two parties however, according to Brezillon [15], has been largely ignored mainly because researchers have focused on issues affecting the architectures for generating explanations, exploring explanatory dialogues and complex architectures as a means to produce sophisticated explanations. Furthermore, Darlington [11] identifies the following areas as distinct research areas in explanations: *Access Mechanism, Types of Explanation Used, Explanation Orientation, the level of Expertise of the User and benefits of explanations*.

We have already discussed types of explanations in the introduction section, however it is important to note that these explanations fall under two categories: User-Invoked explanations which are generated at the request of the user and automatic explanations which are explanations that are given at the discretion of the system [13]. According to Moffitt [20], user-invoked explanations are perceived as a separate component of the KBS which in turn influence users to believe that there is little to no value acquired from explanations. On the other hand, automatic explanations are perceived to be natural to the system.

Darlington [11] states that the orientation of an explanation may either lean towards giving advice on conclusions of the system -called feedback explanation- or advice before the system has reached a conclusion -called feedforward explanations. Feedforward explanations contain terminological explanations which have been discussed in the introduction whereas feedback explanations include explanations about the system's behaviour, justifications, preferences and domain explanations. According to Darlington [11], very little research has been done on the orientation of explanations. Furthermore, he claims that studies conducted on the level of expertise focus on expert level explanations and novice level explanations.

3 OVERVIEW OF EARLY SYSTEMS AND THEIR DRAWBACKS

Currently, KBSs are mainly focused on providing explanations for queries of the type "why" and "how" [23]. Queries of type "how" require an explanation about the the system's behaviour, for example: "How did the the system reach its conclusion?". In order to implement an explanation facility that answers the "how" question, the KBS must explicate its reasoning step by step. On the other hand, queries of the type "why" requires the system to describe and associate its objectives with every calculation it makes.

3.1 Canned Text

The canned text approach to explanation is a rigid approach in which the engineers of the KBS anticipate all possible user queries and hard-code the explanations of each possible answer. As a result, for each parameter that requires a value, there exists an explanation in the form of natural language text that serves as a template explanation.

Advantages and Disadvantages.

This approach worked really well for early KBSs since canned text approaches are easy to prepare and easily used were they are required [23]. The simplicity of this approach allows the explanations to be tailored to various contexts. However, this approach has two serious drawbacks: *a)* the KBS must anticipate all possible queries, *b)* when the KBS is updated, the explanation facilities must be updated too. Lastly, according to Mukundan et al. [23], the canned text approach explanations are unsatisfactory because: "there is no clear link between the explanations and the behaviour that is being explained".

3.2 Code Translation

This approach is context-specific in a sense that the explanations which it produces are strictly depended on the system behaviour. The KBS keeps track of the actions it makes and if a user requires an explanation, it transforms the set of actions it took into a natural language form [23].

Advantages and Disadvantages

The main advantage of this approach over the canned text approach is that it does not have to anticipate all possible user queries and it is fairly easy to implement. Unfortunately, even with this approach, the system does not have an understanding of the actions it takes thus difficult questions may be answered with an undesired level of detail such as a long chain of explanations that consists of a multitude of irrelevant details.

4 EXPLANATIONS IN RECOMMENDER SYSTEMS

A recommender system is an internet-based KBS built with the purpose of helping e-commerce customers find and purchase online products[24]. According to Nava Tintarev and Masthoff [31], a good explanation provided by a recommender system must be:

- **Transparent**: Provide explanations to how it arrived at a particular conclusion (i.e. why did it make a specific recommendation).
- **Scrutable**: All users must be able to indicate to the Recommender System that it has made incorrect decisions.
- **Trustworthy**: Users should be able to rely on the system with a high degree of confidence and trust that the system is acting on the behalf of their interests.
- **Effective**: Help users make optimal decisions. This includes helping users find quality products.
- **Persuasive**: Convince users to try or buy online products.
- **Efficient**: Help users to make decisions as fast as possible.
- **Satisfactory**: Increase the ease of usability and the pleasure of using Recommender Systems.

4.1 Approaches

Based on the handbook published by Ricci et al. [25] in 2010, there are two main approaches to building Recommender Systems: Collaborative filtering (CF) and Content-based filtering. The CF approach is based on belief that customers with similar tastes are most likely to make similar purchasing decisions. For example, if two users have a preference for a certain model of car, it is most likely that they will purchase similar items related to that car. On the other hand Content-based filtering is solely based on the user's profile and the data about the product. The product's description and the user's rating of a small set of products is used to create an estimate about the user's interests which in turn is used to recommend products to the user [33]. For example, if a user has bought a multitude of sports products of a specific brand, it is most likely that s/he will buy another sports product from the same brand.

4.2 Examples of Recommender Systems

LIBRA (Learning Intelligent Book Recommending Agent) is a Recommender System that has a database of approximately 40,000 books in which each book is described using the following details: Author, Titles, Description, Subject, Related Authors and Related Title [4]. It regarded as the first content-based Recommender System that infuses Bayesian learning algorithms and a database of books found from the internet to form recommendations [21]. The main aim of this system is provide users with effective recommendations which help them make good decisions [31]. For example, the system will assess the types of books a user has read and then will recommend books that are useful to the user. The explanation facility of this system is a hybrid of both content-based explanations and collaborative based explanations.

Bilgic and Mooney [4] measure the effectiveness of this system through the following two approaches: the promotion approach and the satisfaction approach. The Promotion approach states that the best explanation is the one that is capable of convincing customers to buy certain products. Whereas, the satisfaction approach states that the best explanation is the one that allows the user to assess the quality of a product in the best possible way.

Furthermore, LIBRA has three explanation facilities, namely Keyword Style Explanation(KSE), Neighbour Style Explanation(NSE) and Influence Style Explanation (ISE).

The KSE (expample shown in *figure 1*)analyses the data of the

Slot	Word	Count	Strength	Explain
DESCRIPTION	HEART	2	94.14	Explain
DESCRIPTION	BEAUTIFUL	1	17.07	Explain
DESCRIPTION	MOTHER	3	11.55	Explain
DESCRIPTION	READ	14	10.63	Explain
DESCRIPTION	STORY	16	9.12	Explain

Figure 1: A figure illustrating The Keyword Style Explanation generated by Bilgic and Mooney [4]

recommended product and attributes a value of strength which is the measure of the likelihood that the product is useful to a user based on the user's profile. Essentially this method answers the question: "what is it about the product that makes it useful to the user" [4].

The NSE (example shown in *figure 2*) facility serves to inform the collaborative component of the Recommender system. It compiles the neighbours' ratings into three categories: Bad, Neutral and Good and then employs the CF technique to generate a recommendation and a visual explanation[4].

Lastly, the ISE provides the user with a table containing products that the user has already rated and has the highest impact on the recommendation of a product.

ACORN (A Conversational Recommender System) is a movie Recommeder System with the primary goals of increasing the trust between the user and the system and persuade user's to rent or buy movies. To achieve its goals, the ACORN system must find a way to capture user preferences and use these preferences to recommend a product[32]. Although, the CF method is adopted by the majority of Recommender Systems, it can only make recommendations based on users' past activities, thus there is a need to combine different recommendation techniques[5]. The ACORN Recommender System is a conversational system therefore it must use some form of dialogue

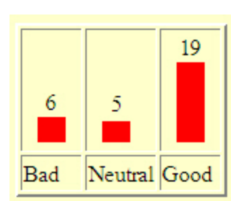

Figure 2: An explanation generated by the NSE facility

to acquire knowledge of user preferences. WÃČÄŘrnestÃČÄŻl [5] points out key aspects of the ACORN system:

- *System-driven preference acquisition*: The system must probe the user for preferences. An example question that the system may ask the user is: “what is your favourite movie?” or “Who is your best actor?”
- *User-driven information requests*: The user must be able to retrieve information as s/he pleases provided that information is contained in the Recommender System. For example, the user may ask “Who is acting in the movie Interstellar?” or may request new movie recommendations.
- *Recommendations in a dialogue context*: The recommendations must be supplied to the user in the form of a dialogue.

Given that this system is a primarily focused on persuasiveness and satisfaction, then its explanation is good if it convinces the user to watch a movie and/or if it does not frustrate the user by giving explanations that are not phrased in a conversational tone (useless responses). Consider the following example of a typical conversation between a user U and the ACORN system A:

A1: How can I help you today?

U1: What action movies would you recommend?
A2: Have you seen ’The Avengers: Endgame’?
U2: No, why?
A3: It is popular among your friends and it generated over a billion dollars in box-office sales.

Note that A3 is an explanation about why the user should watch the movie and its success is dependent on whether the user finds the explanation to be sufficient (i.e. satisfactory) and on whether the user buys the product (i.e. persuasiveness).
Although this paper only focuses on the LIBRA Recommender System and the ACORN System, there is a multitude of Recommender Systems with varying explanation facilities. Each system prioritizes a different property of explanation (i.e. Transparency, Scrutability, Trustworthiness, Effectiveness and Satisfaction) as shown and summarized in *Table 1*.

Table 1: A summary of different Recommender System properties

System	Aim	Product Type	Explanation Facility
LIBRA [21]	Effectiveness	Books	Content-based, Collaborative-based
News Dude [10]	Scrutability, Effectiveness	News	Preference-based
MovieLens [7]	Persuasion, Effectiveness Satisfaction	Movies	Preference-based
MYCIN [31]	Transparency, Effectiveness	Prescriptions	Preference-based
Qwickshop [5]	Effectiveness, Effeciency	Digital Cameras	Preference-based
SASY [9]	Scrutability, Effeciency	Holiday	Preference-based
Adaptive Place Advisor [30]	Effectiveness, Efficiency	Restaurants	Preference-based
Top Case [18]	Transparency, Efficiency	Holiday	Preference-based
Sim [16]	Efficiency	PCs	Preference-based

5 EXPLANATION IN SYMBOLIC KNOWLEDGE BASED SYSTEMS

Symbolic KBS are all systems that require the manipulation of stored symbols in order to make new conclusions. Since the symbols are explicitly stored, it is easy to create explanation facilities for symbolic KBSs. Darlington [11] classifies symbolic KBSs into two categories: model-based KBSs and non-model based KBSs. He classifies KBSs that are amenable to feedforward explanations as model-based KBSs (e.g. rule-based systems and object-oriented systems) and he classifies those that are not well suited to feedforward explanations as non-model based KBSs (e.g. Case-based reasoning).

5.1 Rule-based System

Rule-based systems are KBSs consisting of rules that act and manipulate stored symbolic data. It is easier to build explanation facilities for rule-based systems since they produce a rule-trace (i.e. a procedural description of the steps taken during a calculation) [11]. An example of a Rule-based System is the Digitalis Therapy advisor [28] in which the rule-trace is simply the history of its execution. Feedforward explanations may be generated by this system if the users asks “why a certain conclusion was made”, this is primarily done by tracing the actions of the system and identifying the goals of the system as it take some actions. On the other hand feedback explanations are generated by the Rule-based KBS when a user asks “how did the system reach a conclusion”, this is done by tracing the actions of the system and translating each action of the user into a natural language [11] equivalent statement. The primary advantage of this system is that its explanations reflect the underlying code directly.
On the other hand, this kind of system is more than likely to experience issues of understandability. If the Knowledge Base of the system is not constructed with care, then the resulting explanations may be overwhelmed with unnecessary details such as multiple premises per rule. In such cases the resulting transparency of the system is no better than a black-box reasoning system which is exactly what we are trying to avoid [29]. According to Darlington [11], to reduce the likelihood of the aforementioned disadvantages, procedures (rules) should be broken down into small explainable steps or this can be also fixed by redesigning the knowledge base and procedures of the system. It should be noted that this system cannot form justification explanations if the knowledge base contains incomplete information, for example if the knowledge base only contains premises of the form “A and B imply C”, however you try to reconstruct the knowledge base and rule system, there is no way of generating the explanation for why this premise exists.

5.2 Object Oriented or Frame-based System

This system represents data in a hierarchical manner such that objects are a representation of real world elements (which can be either abstract such as mathematical objects or concrete such as a car) and they can store both data and procedures[19]. The hierarchical nature of this representation allows objects to be related by inheritance. According to Shortliffe et al. [26], this system is able to capture and represent all types of explanations.

5.3 Case-Based Reasoning System

Case-based reasoning system (CBR) is a system that solves problems by using other previous solutions. It is a four-step process[1]:

- **Retrieve**: The first step is to retrieve previous similar cases from memory. This case must be a solution to a problem similar to the one we intend to solve.

  Reuse: Modify the solutions from previous cases such that they can be mapped to the target problem.
- **Revise**: Test the solution to the new problem and make adjustments if required.
- **Retain**: If the new solution successfully solves the target problem, then store the solution in memory as a new case

The description of a retrieved case, also known as a knowledge-light is considered to be an explanation in CBR systems [11]. For instance, according to Cunningham et al. [8], the following knowledge-light qualifies as explanation in a CBR system:

> "The system predicts that the outcome will be X because that was the outcome in case C1 that differed from the current case only in the value of feature F which was f2 instead of f1. In addition the outcome in C2 was also X ..."

Details in an explanation increase at a low level of abstraction. Thus two similar cases may produce radically different explanations when considered from a low level abstraction perspective [12]. On the other hand, when you increase the level of abstraction, at some point the two explanations may become identical since the details that make them distinct are abstracted away.
This also means that justification explanations are are difficult to retrieve for novice users since distinct explanations requires use of low levels of abstractions and novice users may not have the required knowledge to understand the explanations[11]. Although there is very little research comparing explanations from CBR and explanations from Rule based Systems, Cunningham et al. [8] have shown that explanations from CBR systems are preferred over the ones obtained from Rule-based Systems.

6 EXPLANATION IN NON SYMBOLIC SYSTEMS

6.1 Neural Networks

Recall from the last section that Symbolic Systems contain a Knowledge Base in which symbolic premises are explicitly stored. In this case, non symbolic systems are systems in which the Knowledge Base does not explicitly store symbols. Non symbolic systems such as neural networks are notorious for their black-box behaviour. Neural Networks are defined as a directed graph that have values called weights assigned to its edges, a set of nodes which receive input values, a set of nodes that output the target value. Their nodes contain a function called an activation function which is applied when values propagate through them.
Although neural networks have been demonstrated to be successful as classification systems, function approximation systems and Data Processing Systems, their acceptability is obstructed by their inability to explain their actions [3]. Although there is slow progress in making neural networks explainable, supplementing Neural Networks with Symbolic KBS seems to be able to generate explanations. For example, a combination of genetic algorithms and Neural Networks is able to generate explanations albeit with a lower degree of user confidence when compared to Symbolic KBSs [6]. Another example is the decompositional approach proposed by Bader et al. [2] in which explanations are extracted from Neural Networks by supplementing them with a Rule-Based System.

7 CONCLUSIONS

In this paper we explored different kinds of explanations and how they are applied in different kinds of Reasoning Systems. It is observed that the justification explanation is common among the different types of Reasoning System. We learned that one of the reasons why research in explanation facilities is progressing slowly is KBS engineers are concerned about the generation of explanation whereas typical users of such KBSs are focused on the usability of the explanation.
We noticed that early approaches to explanation generation such as canned-text/templates and code translations are not capable of adequately explaining the decisions made by a wide variety of Reasoning Systems. This stems from the fact that canned-text/templates were designed for Rule-based systems and are insensitive to change in the types of Reasoning Systems.

We also observed that Recommender Systems strive for different and specific properties (at most three) of good explanations and this influences the type of experience a user has when interacting with a Recommender System.
We learned that different Symbolic Reasoning Systems generate different explanations whereas the Object-Oriented Symbolic Reasoning is capable of generating all types of explanations identified in this paper.
Another observation is that Non Symbolic Reasoning Systems such as Neural Networks seem to be intrinsically unable to generate explanations unless supplemented with other Symbolic Reasoning Systems.
Besides Non Symbolic Reasoning Systems, all Reasoning Systems tend to evolve towards sophisticated generation of explanations. Finally, Human Computer Interaction has an influence on the value of explanations in a sense that if the system provides explanations that are not tailored to the user's needs then the user will not accept the explanation.

REFERENCES

[1] Agnar Aamodt and Enric Plaza. 1994. Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Commun.* 7, 1 (March 1994), 39–59. http://dl.acm.org/citation.cfm?id=196108.196115

[2] Sebastian Bader, Steffen HÃűlldobler, and Valentin Mayer-Eichberger. 2007. Extracting Propositional Rules from Feed-forward Neural Networks — A New Decompositional Approach, Vol. 230.

[3] Bart Baesens, Rudy Setiono, Christophe Mues, and Jan Vanthienen. 2003. Using Neural Network Rule Extraction and Decision Tables for Credit-Risk Evaluation. *Management Science* 49 (03 2003), 312–329. https://doi.org/10.1287/mnsc.49.3.312.12739

[4] Mustafa Bilgic and Raymond Mooney. 2005. Explaining Recommendations: Satisfaction vs. Promotion.

[5] Robin Burke. 2002. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* 12 (11 2002). https://doi.org/10.1023/A:1021240730564

[6] Russell C. Eberhart. 1992. The role of genetic algorithms in neural network query-based learning and explanation facilities. 169 – 183. https://doi.org/10.1109/COGANN.1992.273940

[7] Dan Cosley, Shyong K. Lam, Istvan Albert, Joseph A. Konstan, and John Riedl. 2003. Is Seeing Believing?: How Recommender System Interfaces Affect Users' Opinions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, New York, NY, USA, 585–592. https://doi.org/10.1145/642611.642713

[8] Padraig Cunningham, Donal Doyle, and John Loughrey. 2003. An Evaluation of the Usefulness of Case-Based Explanation. In *In Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer, 122–130.

[9] Marek Czarkowski and Judy Kay. 2002. A Scrutable Adaptive Hypertext, Vol. 2347. 384–387. https://doi.org/10.1007/3-540-47952-X_43

[10] Michael J. Pazzani Daniel Billsus. 1999. A Personal News Agent that Talks, Learns and Explains. *Third International Conference on Autonomous Agents* (1999).

[11] Keith Darlington. 2013. Aspects of Intelligent Systems Explanation. *Universal Journal of Control and Automation,* 1 (10 2013), 40–51. https://doi.org/10.13189/ujca.2013.010204.

[12] DÃşnal Doyle, Alexey Tsymbal, and Padraig Cunningham. 2019. A Review of Explanation and Explanation in Case-Based Reasoning. (04 2019).

[13] Shirley Gregor and Izak Benbasat. 1999. Explanations From Intelligent Systems: Theoretical Foundations and Implications for Practice. *MIS Quarterly* 23 (12 1999), 497–530. https://doi.org/10.2307/249487

[14] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. 2018. Local Rule-Based Explanations of Black Box Decision Systems. *CoRR* abs/1805.10820 (2018). arXiv:1805.10820 http://arxiv.org/abs/1805.10820

[15] Sebai Hakima and Franz Oppacher. 1990. Improving explanations in knowledge-based systems: RATIONALE. *Knowledge Acquisition* 2 (12 1990), 301–343. https://doi.org/10.1016/S1042-8143(05)80012-3

[16] Lorraine Mcginty and Barry Smyth. 2019. Extending Comparison-Based Recommendation: A Review. (05 2019).

[17] Deborah L. McGuinness and Peter F. Patel-Schneider. 1998. Usability Issues in Knowledge Representation Systems. In *AAAI/IAAI*.

[18] David Mcsherry. 2005. Explanation in Recommender Systems. *Artificial Intelligence Review* 24 (10 2005), 179–197. https://doi.org/10.1007/s10462-005-4612-x

[19] M Minsky. 1975. A Framework for Representing Knowledge. *P.H. Winston (ed.), The Psychology of Computer Vision, McGraw-Hill, New York, pp. 211-277* (01 1975).

[20] Kathleen Ellen Moffitt. 1989. *An Empirical Test of Expert System Explanation Facility Effects on Incidental Learning and Decision-making*. Ph.D. Dissertation. Tempe, AZ, USA. AAI9018520.

[21] Raymond J. Mooney and Loriene Roy. 2000. Content-based Book Recommending Using Learning for Text Categorization. In *Proceedings of the Fifth ACM Conference on Digital Libraries (DL '00)*. ACM, New York, NY, USA, 195–204. https://doi.org/10.1145/336597.336662

[22] Johanna Moore and William Swartout. 1989. Explanation in expert systems: A survey. (01 1989).

[23] Sasikumar Mukundan, Srinivasan Ramani, S Muthu Raman, KSR Anjaneyulu, and Raman Chandrasekar. 2007. A Practical Introduction to Rule Based Expert Systems. (01 2007).

[24] Francesco Ricci, D.R. Fesenmaier, Nader Mirzadeh, Hildegard Rumetshofer, Erwin Schaumlechner, Adriano Venturini, Karl WÃűber, and Andreas Zins. 2006. 4 DieToRecs: A Case-based Travel Advisory System. (07 2006).

[25] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. 2010. *Recommender Systems Handbook* (1st ed.). Springer-Verlag, Berlin, Heidelberg.

[26] Janice S. Aikins, John C. Kunz, Edward Shortliffe, and Robert J. Fallat. 1983. PUFF: An expert system for interpretation of pulmonary function data. *Computers and biomedical research, an international journal* 16 (07 1983), 199–208. https://doi.org/10.1016/0010-4809(83)90021-6

[27] William Swartout and Johanna Moore. 1993. Explanation in Second Generation Expert Systems. 543–585. https://doi.org/10.1007/978-3-642-77927-5_24

[28] W. R. Swartout. 1977. *A DIGITALIS THERAPY ADVISOR WITH EXPLANATIONS*. Technical Report. Cambridge, MA, USA.

[29] William R. Swartout. 1983. XPLAIN: a system for creating and explaining expert consulting programs. *Artificial Intelligence* 21, 3 (1983), 285 – 325. https://doi.org/10.1016/S0004-3702(83)80014-9

[30] Cynthia A. Thompson, Mehmet H. Göker, and Pat Langley. 2004. A Personalized System for Conversational Recommendations. *J. Artif. Int. Res.* 21, 1 (March 2004), 393–428. http://dl.acm.org/citation.cfm?id=1622467.1622479

[31] Nava Tintarev and Judith Masthoff. 2007. A Survey of Explanations in Recommender Systems. In *Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering Workshop (ICDEW '07)*. IEEE Computer Society, Washington, DC, USA, 801–810. https://doi.org/10.1109/ICDEW.2007.4401070

[32] Pontus WÃďrnestÃěl. 2019. User evaluation of a conversational recommender system. (05 2019).

[33] Harry Zisopoulos, Savvas Karagiannidis, Georgios Demirtsoglou, and Stefanos Antaris. 2008. Content-Based Recommendation Systems. (11 2008).