Eliciting Inductive User Preferences for Multimedia Information Retrieval

David Zellhöfer
BTU Cottbus
Postfach 10 13 44
03013 Cottbus
david.zellhoefer@tu-cottbus.de

ABSTRACT
Preferences have gained a tremendous impact on the personalization of user queries. In this paper, we present an approach for preference elicitation in a multimedia information retrieval scenario. The approach is based on inductive preferences, which provide an intuitive means for stating preferences on actual result documents. Additionally, they do not demand further knowledge of the underlying retrieval system of the user. This work focuses on the user interaction of preference formulation and presents a prototype for a preference-based multimedia information retrieval system.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance Feedback*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Theory and Methods, User-centered Design*; H.2.4 [**Database Management**]: Systems—*Query Processing*

General Terms
User Preferences, Relevance Feedback, User Interface, Condition Weighting, DB & IR, Machine-based Learning

1. INTRODUCTION
Over the past years, preferences gained a tremendous impact on the personalization of user queries. Originally having a strong background in economics [2], they have become a vividly discussed topic in computer science – especially in databases (DB) [5, 11] or in artificial intelligence [4].
In the domain of (multimedia) information retrieval (IR), which is the main focus of this paper, personalization is an important factor to improve user satisfaction. Means of personalization range from relevance feedback (RF) [16] to weighting approaches, e.g., in extended boolean retrieval models [13].
Because traditional approaches for IR that are based on a combination of low-level features such as histograms or the like are currently hitting a "glass ceiling" [1], personalization is considered the last important contributor to an improved retrieval performance.
In this paper, we will present an approach for preference elicitation in multimedia retrieval using *inductive preferences*. This paper will focus on one aspect of the retrieval process: user-defined preferences and their interactive modification.

The remainder of the paper is structured as follows. In the next section, related work will be discussed. Sec. 2 will present our approach in detail and presents a prototypical user interface. The paper ends with a conclusion including issues that we consider important and that we will address during future research.

1.1 Related Work
Preferences in DB can be subsumed under two categories: qualitative [5, 11] and quantitative [7, 19] approaches.
Qualitative approaches use partially ordered sets (posets) in order to model preference queries. Skyline queries [3, 12] are part of this class. In general, qualitative approaches can be considered intuitive because they allow users to state preferences in a manner such as: "I like object A better than B". In addition, they are fully compatible with the relational model. As a consequence, results of such approaches form sets without an intra-ordering of the elements and *ceteris paribus* semantics.
This set semantic contrasts with IR. Here, users usually expect a ranking, i.e., a total order of result objects. This ranking is ordered by the "probability of relevance" of result documents w.r.t. a given query (document). *Quantitative approaches* offer this characteristic and are common in IR [17, 23, 18]. In the DB domain, quantitative approaches are known as well and rely on numeric weights on logical connected conditions in order to express user preferences [7]. Setting these weights is a complicated task and often hidden from users because of its cognitive burden.

Regarding multimedia IR in particular, another issue has to be addressed. Multimedia documents can be described with different representations during the retrieval process, e.g., low-level and high-level features such as annotations and meta data like date of creation, media type or the like. While the latter can be easily stored within a (relational) DB, the first representations are usually accessed using traditional IR techniques. Hence, [19] proposed a framework combining IR- and DB-like queries. This theoretical framework manifests in *CQQL*, the commuting query language, which forms the basis for the preference model discussed through-

out this paper. CQQL contributes to the quantitative approaches. Atop of this query language, inductive preferences will be used for user interaction. Inductive preferences (see Sec. 2) have a qualitative characteristic, which links them to *example-critiquing* [22]. These preferences serve as input for a learning algorithm that generates a CQQL query fitting the specified preferences.
Thus, the presented work is also related to [27], which deals with formal characteristics and the specification of the utilized learning algorithm that will be used within this paper.

2. USER MANIPULATION OF INDUCTIVE PREFERENCE QUERIES

As said before, setting weights within a quantitative query language imposes a major burden on users. This is due to three main reasons:

1. All queried attributes have to be known to the user, i.e., their intrinsic semantics have to be comprehensible.
2. The interaction between these attributes and the user's preference amongst them has to be known.
3. The preference has to be expressed with numeric values by the user.

Qualitative approaches suffer also from the first two problems, which is due to their deductive nature. This becomes obvious if trade-offs for skylines [15, 14] are considered. Here, trade-offs are specified between certain attributes resulting in a relaxation of the skyline. To state a trade-off, the semantics of an attribute has to be known to the user in order to be able to express a preference such as a green car is preferred to a red one.
Unfortunately, attributes in multimedia IR are more complex. Commonly used attributes representing multimedia documents rely on various kinds of low-level features such as histograms, spectrograms or correlograms, high-level annotations in natural language and meta data. Note that natural language will be represented internally with the means of IR such as the vector space model [18] that are not necessarily generating comprehensible similarity values of a document w.r.t. a query. It can be argued if the latter, the meta data, is comprehensible when stored within a DB. Nevertheless, even if all attributes could be understood by the user it can be doubted if a clear preference amongst an arbitrary combination of them can be formulated easily. While this problem is not this severe in qualitative approaches because relations between attributes like "better-than" can be formulated, quantitative approaches need the user to express these preferences on a numerical basis between the attributes.
To overcome these problems, our approach uses a hybrid method combining both qualitative and quantitative preference paradigms.

2.1 Inductive Preferences

In order to offer an intuitive means for preference elicitation that can be used in multimedia IR scenarios, we introduced the concept of *inductive preferences* [25]. Their name is due to the fact that we derive a deductive and quantitative CQQL query from qualitative preferences between arbitrary result documents using inductive reasoning. The inductive reasoning is carried out by a machine-based learning algorithm that finds weights for a given CQQL query that express the user-specified preferences. For a detailed description of the algorithm see [27]. The direct specification of qualitative preferences on actual result objects to personalize the query has two major advantages:

1. Stating qualitative preferences is intuitive. Given two objects, it is easy for a user to state a preference such as "I like A better than B".
2. Preference decisions between result documents without complete knowledge of all underlying facts is known from daily life. In our scenario this relieves the user from a confrontation with the document's attributes that are used by the retrieval system, thus, lowering the cognitive burden.

This way of eliciting preferences is related to *example-critiquing* [22], which utilizes a similar mechanism. In contrast to this approach, inductive preferences are used to learn weights for a CQQL query, i.e., a logical deductive query using boolean operators. The usage of such a structured query improves the retrieval performance, which has been shown in IR [10, 21]. [22] address preferences on attributes alone that – additionally – must be comprehensible to the user because a decision tree-like mechanism is used for preference refinement.

Fig. 1 illustrates the embedding of the aforementioned inductive preference elicitation into an iterative RF process. Given a weighted CQQL query, an initial ranking R_0 is generated[1]. Based on this ranking, the user can state inductive preferences between sample documents ultimately creating a preference poset P_1. An inductive preference $p_{i,j}$ between two documents in a document collection D is defined as an element of a binary ordering relation $R \subseteq D \times D$, e.g. $d_i \geq d_j$ whereas $d_i, d_j \in D$. In a following learning step, this poset is used to derive new weights for the CQQL query fulfilling the preferences. This process is iterated until the user is satisfied with the results. See [27] for a detailed explanation.

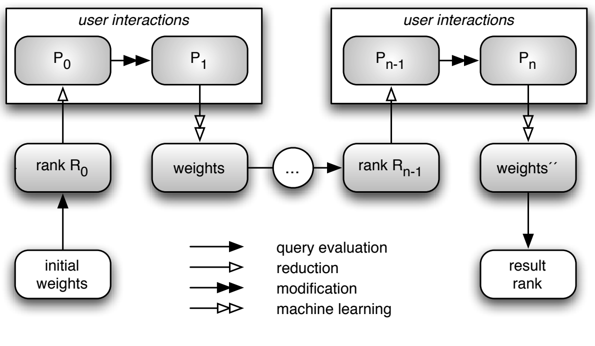

Figure 1: Refinement process for preferences [27]

[1]For a discussion of possible initial weighting schemes see [26], although further knowledge is not needed to understand this paper.

2.2 Conflict Detection

Although inductive preferences are an intuitive means for preference elicitation, they contain the risk of including conflicting assertions. This is mainly due to the free user interaction with the result documents in an iterative manner. To clarify this, we have to consider the process of preference specification from a graph-theoretical point of view.
As said before, all preferences eventually form a poset, which serves then as input for the learning algorithm. The poset can be regarded as a directed graph with nodes representing documents and edges depicting preference relations between these documents, i.e., we obtain a Hasse diagram if no conflicting assertions have been made. In order to detect conflicting assertions introduced by the user, we have to focus on one major characteristic of a Hasse diagram: its acyclicity. If the graph is not acyclic it can be shown that a *conflict* has been specified, i.e., a poset of preferences such as $d_1 > d_4 > d_2 > d_1; d_2 > d_5; d_6 > d_3$ (Fig. 2). In other words, cycles being present in the graph mirror violations of antisymmetry and transitivity, which are required for a poset. The detection of cycles within a graph is an old problem in computer science [8] and can be solved by a topological sorting of the graph. Although topological sorting has a bad complexity class ($\mathcal{O}(n^2)$, average $\mathcal{O}(n)$), it is likely that the graph will not contain a lot of nodes because users tend to avoid excessive interaction with systems [20]. In addition, the size of the result is limited, thus restricting the amount of possible preferences.

2.3 Conflict Resolution

Because of the iterative nature of the presented approach, preferences will be added, modified, or removed by the user during each step. That means that the preference graph has to be checked after every user interaction. While it may seem tempting to check the graph within fixed intervals, this will lead to usability issues. If we check the graph after every, e.g., 10 user interactions, the user may have created 10 new preferences. These preferences will be included into the old poset P_i using the set union operator, which will ignore possible conflicts [6]. If we now show the conflicts to the user she will have to decide how to resolve them. Unfortunately, there will be different possibilities to remove the cycles of which some will introduce new cycles. In consequence, the actual removal will become a tedious and error-prone task. Hence, we check for conflicts parallel to the user interaction with the system in order in support users with the reversal of their actions and to prevent additional errors as demanded of the golden rules of interface design [20].

In order to help the user with the conflict resolution, some conflicts can be resolved automatically. The automatic resolution of conflicts has been studied for qualitative approaches [15, 14] and focuses mainly on the prioritization of some attributes, i.e., the definition of a trade-off. As said before, trade-offs rely on the comprehensibility of attributes, which can not be guaranteed in a multimedia IR system. Hence, these contributions cannot be used directly in the presented scenario.
Our approach features two conflict resolution techniques of which one is automatic and one is semi-automatic. The simplest automatic technique for conflict resolution is the *prioritization* of one conflicting preference over another, i.e., if a conflict has been discovered, the user has to decide between two preferences (Fig. 3, a or b). The preferences will be shown to the user based on the sample documents that were chosen (c). This approach can be executed fully automatic if the system decides, which preference has to be prioritized. This conclusion is based on the user interaction, e.g., which preference has been chosen first or how long a user has looked at a certain document being involved in a conflict preference. In order to determine, which technique is more consistent with the user expectations, user studies have to be conducted. Alternatively, [6] suggested a Pareto composition of preferences. This conflict resolution technique can be carried out automatically.

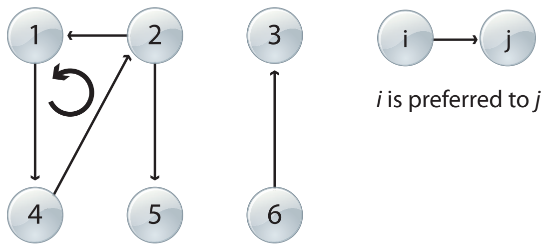

Figure 2: Hasse diagram of preferences including a conflict between document 2 and document 4

2.4 Prototypical User Interface

In order to test the utility of our approach, we are currently working on a graphical user interface (GUI) for preference elicitation in an image retrieval system. As said before, the preference poset can be visualized as a Hasse diagram, which has been done, e.g., by [24]. Nevertheless, using a Hasse diagram is not feasible as a means of interaction in a non-expert system. This is due to two main reasons:

1. Hasse diagrams can become complex, which increases the cognitive workload for a users because of the sheer amount of nodes and edges that are present for a decent number of preferences lowering the visual separability of preferences.

2. The modification of a preference in a Hasse diagram is a error-prone and cognitively demanding task.

To clarify the second issue, consider the following example. The modification of preferences within a Hasse diagram involves the creation of new edges or nodes. Hence, the user is likely to introduce cycles into the diagram, which are forbidden and have to be communicated. In order to modify preferences without introducing new cycles such interfaces require users to stress their short-term memory and act consistently in order to prevent or foresee errors. This poses a tremendous cognitive workload on users eventually lowering the overall usability of such a system.
For the first prototype, we decided to implement a user interface that prevents the input of conflicts during user interaction. In our approach, preferences can be modeled using "concentric circles". Here, the query document is depicted in the center of the concentric circles (Fig. 4, right) while preferred documents are arranged around this center. Every ring serves as a preference level, i.e., a document on an inner ring is preferred to a document on an outer ring. Documents

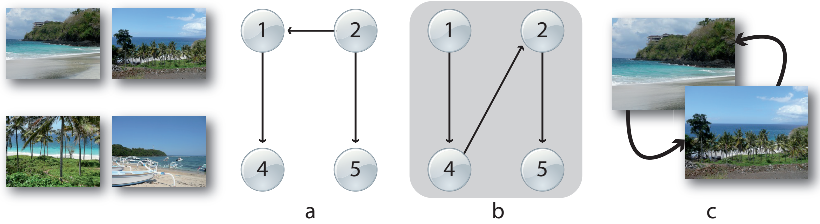

Figure 3: Conflict resolution by prioritized composition on basis of user interaction with the document collection (based on the conflict in Fig. 2)

that will participate in an inductive preference are simply dragged from the result list (Fig. 4, left) to the preference area (right). In order to prevent conflicts, the prototype allows documents being present only once in the preference area. Therefore, cycles cannot occur. In addition to fine granular inductive preferences, the prototype is also capable of dealing with traditional binary *relevance/irrelevance* judgments in RF that are widely used in IR [9]. In fact, the presented poset approach forms a generalization of such judgments. Binary judgments can be modeled by considering all relevant documents as the upper bound of the poset and all irrelevant as the lower bound.

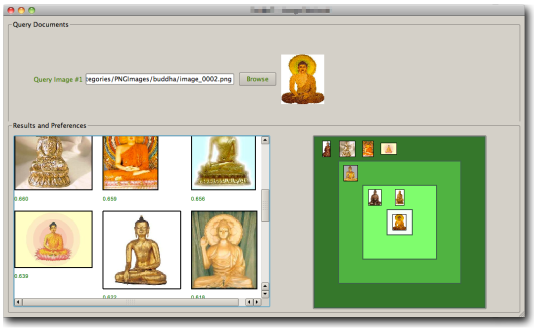

Figure 4: Prototype with "concentric circles" for preference elicitation

3. CONCLUSIONS AND FUTURE WORK

This contribution combines strengths from qualitative and quantitative preference approaches and discusses its application in multimedia information retrieval. It amalgamates qualitative *inductive preferences* that are known from daily life with a quantitative query language using a machine-based learning algorithm. These inductive preferences can be modified by users interactively and directly on result documents. Because this can lead to conflicting preference assertions, methods for conflict resolution are presented and evaluated.

The presented approach is embedded into a relevance feedback mechanism that has been implemented in a first graphical prototype. The current prototype addresses layperson users and provides a simple means to elicit preferences in an error-preventing and simple way using concentric circles. Hence, it integrates the user-friendly notion of inductive preferences with a ranking model that is commonly used in IR.

Fundamentally, some additional issues need further investigation. We plan to conduct user tests to gain reliable insights about expectations w.r.t. automatic conflict resolution techniques. Regarding the learning algorithm, first results show that it can be extended to formula learning. To conclude with, our main goal is to develop a user-centered and supportive multimedia IR system utilizing a RF mechanism without increasing the user's cognitive workload.

4. ACKNOWLEDGMENTS

The author would like to thank all students that have contributed to the current work, especially Bianca Böckelmann for the latest work with the GUI prototype. He thanks Ingo Schmitt for the valuable discussions and the support of his research.

5. REFERENCES

[1] J. J. Aucouturier and F. Pachet. Improving Timbre Similarity: How high is the sky? In *Journal of Negative Results in Speech and Audio Sciences*, volume 1 of *1*. 2004.

[2] S. G. Becker. *The economic approach to human behavior*. Univ. of Chicago Press, Chicago, paperback ed., [nachdr.] edition, 2008.

[3] S. Börzsönyi, D. Kossmann, and K. Stocker. The Skyline Operator. In *Proceedings of the 17th International Conference on Data Engineering*, pages 421–430. IEEE Computer Society, 2001.

[4] C. Boutilier, I. R. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements. *Journal of Artificial Intelligence Research*, 21:2004, 2003.

[5] J. Chomicki. Querying with Intrinsic Preferences. In *EDBT '02: Proceedings of the 8th International Conference on Extending Database Technology*, pages 34–51. Springer-Verlag, 2002.

[6] J. Chomicki. Database querying under changing preferences. *Annals of Mathematics and Artificial Intelligence*, 50(1–2):79–109, 2007.

[7] R. Fagin and L. E. Wimmers. A Formula for Incorporating Weights into Scoring Rules. *Special Issue of Theoretical Computer Science*, (239):309–338, 2000.

[8] C. R. Holt. Some Deadlock Properties of Computer Systems. *ACM Computing Surveys*, 4(3):179–196, 1972.

[9] J. M. Huiskes and S. M. Lew. Performance evaluation of relevance feedback methods. In *CIVR '08:*

Proceedings of the 2008 international conference on Content-based image and video retrieval, pages 239–248. ACM, 2008.

[10] A. D. Hull. Using Structured Queries for Disambiguation in Cross-Language Information Retrieval. In *AAAI Spring Symposium on Cross-Language Text and Speech Retrieval Electronic Working Notes*, pages 24–26, 1997.

[11] W. Kießling. Foundations of preferences in database systems. In *VLDB '02: Proceedings of the 28th international conference on Very Large Data Bases*, pages 311–322. VLDB Endowment, 2002.

[12] D. Kossmann, F. Ramsak, and S. Rost. Shooting Stars in the Sky: An Online Algorithm for Skyline Queries. In *Proc. of the 28th Int. Conf. on Very Large Data Bases VLDB'02, Hong Kong, China, August, 2002*, pages 275–286. Morgan Kaufmann Publishers, 2002.

[13] H. J. Lee. Properties of Extended Boolean Models in Information Retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 182–190. Springer-Verlag New York, Inc., 1994.

[14] J. Lee, G.-w. You, and S.-w. Hwang. Telescope: Zooming to Interesting Skylines. In K. Ramamohanarao, R. P. Krishna, K. M. Mohania, and E. Nantajeewarawat, editors, *Advances in Databases: Concepts, Systems and Applications, 12th International Conference on Database Systems for Advanced Applications, DASFAA 2007, Bangkok, Thailand, April 9-12, 2007, Proceedings*, volume 4443 of *Lecture Notes in Computer Science*, pages 539–550. Springer, 2007.

[15] C. Lofi, W.-T. Balke, and U. Güntzer. Consistency Check Algorithms for Multi-Dimensional Preference Trade-Offs. *International Journal of Computer Science & Applications (IJCSA)*, 5(3b):165–185, 2008.

[16] J. Rocchio. Relevance Feedback in Information Retrieval. In *The SMART Retrieval System*, pages 313–323. 1971.

[17] G. Salton, A. E. Fox, and H. Wu. Extended Boolean Information Retrieval. *Commun. ACM*, 26(11):1022–1036, 1983.

[18] G. Salton, A. Wong, and S. C. Yang. A Vector Space Model for Automatic Indexing. Ithaca, NY, USA, 1974.

[19] I. Schmitt. QQL: A DB&IR Query Language. *The VLDB Journal*, 17(1):39–56, 2008.

[20] B. Shneiderman and C. Plaisant. *Designing the user interface: Strategies for effective human-computer interaction.* Pearson, Boston, 4. ed. edition, 2005.

[21] H. Turtle and B. W. Croft. Evaluation of an inference network-based retrieval model. *ACM Trans. Inf. Syst.*, 9(3):187–222, 1991.

[22] P. Viappiani, B. Faltings, and P. Pu. Preference-based search using example-critiquing with suggestions. *J. Artif. Int. Res.*, 27(1):465–503, 2006.

[23] W. Waller and H. D. Kraft. A Mathematical Model of a Weighted Boolean Retrieval System. *Information Processing and Management*, 15(5):235–245, 1979.

[24] Wolf-Tilo Balke and Ulrich Güntzer and Christoph Lofi. User Interaction Support for Incremental Refinement of Preference-Based Queries. In Colette Rolland and Oscar Pastor and Jean-Louis Cavarero, editors, *Proceedings of the First International Conference on Research Challenges in Information Science, RCIS 2007, Ouarzazate, Morocco, April 23-26, 2007*, pages 209–220. 2007.

[25] D. Zellhöfer. Inductive User Preference Manipulation for Multimedia Retrieval. In *The Second International Conferences on Advances in Multimedia*, 2010.

[26] D. Zellhöfer and I. Schmitt. A Poset Based Approach for Condition Weighting. In AMR, editor, *6th International Workshop on Adaptive Multimedia Retrieval.* 2008.

[27] D. Zellhöfer and I. Schmitt. A Preference-based Approach for Interactive Weight Learning: Learning Weights within a Logic-Based Query Language. *Distributed and Parallel Databases*, 2009.