How applicable are attribute-based approaches for human-centered ranking creation?

Barth, Clara-Maria ; Schmid, Jenny ; Al-Hazwani, Ibrahim ; Sachdeva, Madhav ; Cibulski, Lena ; Bernard, Jürgen

DOI: https://doi.org/10.1016/j.cag.2023.05.004

Originally published at:
Barth, Clara-Maria; Schmid, Jenny; Al-Hazwani, Ibrahim; Sachdeva, Madhav; Cibulski, Lena; Bernard, Jürgen (2023). How applicable are attribute-based approaches for human-centered ranking creation? Computers Graphics, 114:45-58.
DOI: https://doi.org/10.1016/j.cag.2023.05.004

Computers & Graphics 114 (2023) 45–58

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on EuroVA 2022

How applicable are attribute-based approaches for human-centered ranking creation?

Clara-Maria Barth [a,*], Jenny Schmid [a], Ibrahim Al-Hazwani [a,d], Madhav Sachdeva [a], Lena Cibulski [b,c], Jürgen Bernard [a,d]

[a] University of Zurich, Zurich, 8006, Switzerland
[b] Fraunhofer IGD, Darmstadt, 64283, Germany
[c] Technical University of Darmstadt, Darmstadt, 64289, Germany
[d] Digital Society Initiative, Zurich, 8006, Switzerland

ARTICLE INFO

Article history:
Received 15 October 2022
Received in revised form 23 April 2023
Accepted 10 May 2023
Available online 16 May 2023

Keywords:
Item ranking
Attribute-based ranking
Human-centered ranking creation
Human factors
Visual analytics
Experimental study

ABSTRACT

Item rankings are useful when a decision needs to be made, especially if there are multiple attributes to be considered. However, existing tools do not support both categorical and numerical attributes, require programming expertise for expressing preferences on attributes, do not offer instant feedback, lack flexibility in expressing various types of user preferences, or do not support all mandatory steps in the ranking-creation workflow. In this work, we present RankASco: a human-centered visual analytics approach that supports the interactive and visual creation of rankings. The iterative design process resulted in different visual interfaces that enable users to formalize their preferences based on a taxonomy of attribute scoring functions. RankASco enables broad user groups to (a) select attributes of interest, (b) express preferences on attributes through interactively tailored scoring functions, and (c) analyze and refine item ranking results. We validate RankASco in a user study with 24 participants in comparison to a general purpose tool. We report on commonalities and differences with respect to usefulness and usability and ultimately present three personas that characterize common user behavior in ranking-creation. On the human factors side, we have also identified a series of interesting behavioral variables that have an influence on the task performance and may shape the design of human-centered ranking solutions in the future.

1. Introduction

In everyday life, people constantly face the challenge of finding the best item in an item set: whether it is about picking the nicest hotel for a holiday trip, the next movie to watch, the most promising stock to buy, or the perfect flat to rent. Item sets typically contain large numbers of items to choose from, each of which is defined across multiple attributes representing different criteria to be considered carefully. Such a multi-attribute choice [1] is not an easy task, especially for non-experts. An obvious optimal solution does generally not exist, and "best" highly depends on the decision-maker's personal preferences. Also, the task complexity heavily depends on the dataset size and the number of relevant attributes, both subject to growth.

A strategy to identify items of choice in large item sets is the creation of item rankings. A striking benefit of rankings is the inherent order they provide to items, enabling users to easily find most preferred items at the top. In turn, least preferred items for a decision-making scenario are situated at the bottom of the ranking. We focus on human-centered approaches for the creation of item rankings, leveraging individual *preferences* of users as a profound basis to express multiple criteria to optimize for. Traditionally, many people relied on pen and paper or general purpose spreadsheet tools to formalize and create item rankings. With the digital transformation, people can make use of more sophisticated computational support to ease the creation of item rankings. Still, interactively engaging with the *creation* and *refinement* of item rankings is desirable for everyone: not only for domain experts or users with programming expertise but also for non-experts.

Strategies for interactive ranking creation are two-fold. *Item-based* approaches allow users to express feedback about the perceived order and relevance of data items [2–5]. Users can directly interact with items of interest, make item comparisons, and adjust the ranks of items, e.g., in spreadsheets. *Attribute-based* approaches allow users to express preferences on attributes. Algorithms then transform these preferences into attribute scores, combine the attribute scores according to some weighting, and

[*] Corresponding author.
E-mail address: clara-maria.barth@uzh.ch (C.-M. Barth).

https://doi.org/10.1016/j.cag.2023.05.004

produce a ranking as the direct result of ordering items by their overall scores [6,7]. With the proposal of Attribute Scoring Functions (ASFs) [8], we have presented the formal underpinning to define user preferences on attributes.

In this work, we provide an extension to RankASco [9], a visual analytics tool around ASFs to create attribute-based item rankings. We focus on the attribute-based creation of item rankings for two reasons. First, we believe that its scalability is mainly agnostic to the number of items, making it more applicable for large item sets. Second, we assume that it is easier for users to express preferences for individual attributes than between items as a whole. A pioneer visual analytics approach for multi-attribute ranking is *LineUp* [6]. It offers a visual interface that allows users to map attribute values to preference scores, even if *LineUp* does not offer full flexibility regarding types of user preferences.

The reflection on the body of related work on ranking creation in general revealed five shortcomings. First, existing solutions do not yet offer the flexibility users may require to intuitively express their preferences regarding attribute values. In specific, we identify a lack of tools that support both categorical and numerical ASFs. Second, most existing tools require programming when it comes to ASF creation. These tools can only be steered by math experts or computer scientists, but not by non-experts. Third, the black-box nature of the programming paradigm does not offer instant feedback about the distribution of attribute values (data), how a created ASF behaves (model), or how interactive refinements by users affect the process (user). Fourth, hardly any tool supports all mandatory steps in the ranking workflow: creation, refinement, and usage [8]. Finally, not much is known about the users of ranking creation tools. In particular, a deeper understanding about common ranking creation behaviors could help the design and development of (human-centered) ranking systems in the future. In this context, the evaluation of approaches for the characterization of user groups, such as personas, could be useful to better understand user needs when creating rankings.

To this end, we revisit and extend RankASco [9]. Our contributions are as follows:

- The presentation of RankASco, an attribute-based visual analytics approach that accepts user preferences to create rankings for large item sets. RankASco is the result of a two-year research project, with two workshop paper publications [8, 9] forming the baseline for this extended version. We build upon RankASco with additional visual interfaces, refined design choices, and more descriptive details.
- The validation of RankASco in a user study with 24 participants. The study evaluates RankASco in comparison to Excel as a representative of a general purpose tool. We decided to recruit non-experts with low familiarity in using programmatic solutions to solve multi-criteria ranking problems.
- The presentation of three personas, characterizing common user behaviors in ranking-creation: (1) Peter, the perfectionist, (2) Eva, the explorer, and (3) Pippa, the pragmatist.

By providing visual interfaces for all eight types of attribute scoring functions, our approach is the first that allows users to express a large variety of attribute-based preferences, for categorical and numerical attributes alike. In an iterative design process, we have developed RankASco to make the task of ranking creation accessible to a broad range of users. As a result of careful design, development, and validation, RankASco provides a framework that supports multi-criteria decision-making for the general public. With the identification of three personas for item ranking, we hope to guide the design of future human-centered ranking solutions.

2. Related work

Ranking creation typically relies on algorithmic models that leverage data characteristics to infer an item order. We extend this principle towards human-centered creation of rankings, which encourages users to interactively engage with the underlying data and express preferences on items or attributes. We structure related works along our main contributions: the general role of human preference expression (Section 2.1), the human-centered creation of item rankings (Section 2.2), and the evaluation of interactive approaches for ranking creation and personalization (Section 2.3).

2.1. Expression of user preferences

Providing the users with the ability to input their preferences is a crucial aspect of human-centered design in various fields, such as recommender systems, visual analytics, and human–computer interaction. There are two main approaches to gather user preferences: implicit feedback and explicit feedback. Implicit feedback is based on collecting information about the users' preferences by watching their natural interaction with the systems, e.g., number of clicks or time spent on a page. Explicit feedback requires the users to explicitly express feedback, e.g., by selecting and marking documents and providing ratings for specific items. The main advantage of implicit feedback is that there is no cost for the user to provide feedback. However, it is generally thought that the implicit strategy tends to be less accurate than explicit feedback [10]. For a detailed comparison between these two approaches, please refer to existing studies [11,12].

2.2. Human-centered ranking creation

Ranking creation in real-world settings is mostly performed by third-party platforms, thus leaving users only with the resulting ranking. Most web shops, movie streaming services, and online browsing follow this line of approach. Algorithmic support for ranking creation often involves recommender systems [13,14] or other types of machine learning methods [15–17]. We exclude this branch of approaches, as it does not allow users to explicitly create rankings by themselves, thus not following the human-centered principle. In fact, some third-party approaches enable users to *personalize* existing rankings, but they do not allow for initial ranking creation. In contrast, human-centered ranking creation offers a high degree of human control [18], where they can apply preferences either to items or attributes. This distinction structures our reflection on related works.

Item-based approaches allow users to explicitly express feedback about items and their perceived order to arrive at a personalized ranking. *TasteWeights* [2] enables users to iteratively adjust item preferences using slider widgets. While users can directly observe how their modification of preferences affects the ranking, the approach cannot assign negative item preference scores to indicate disfavor. *RankKit* [4] exploits the users' knowledge at an item level by providing a user-friendly interface for users to manually rank known items. As a beneficial side effect, the authors identified an increase in user trust towards the resulting ranking, which stems from real-time visual feedback on user's interactions. Finally, *Podium* [19] is a multi-attribute approach that enables users to drag items across the ranking to reflect the perceived relative relevance of items. Podium then infers the parametrization of a ranking SVM model to match these preferences. To complement the computational support, users can also change the weights of attributes contributing to the item ranking. Off-the-shelf spreadsheet approaches such as *Microsoft Excel*, *Google Sheet*, and *Apple Numbers* can be seen as item-based approaches, enabling users to perform analysis tasks like filtering and sorting. Manageable task complexity depends on the user's level of expertise: if users are required to solve a complex ranking task, some considerable scripting skills will be required.

Attribute-based approaches allow users to explicitly express preferences regarding specific attributes and attribute values of items. In previous work [8], we studied different approaches that can be used for transforming attribute values into scores, ranging from merely theoretical approaches [20,21] to visual interactive approaches [22–24]. The resulting taxonomy of eight types of attribute scoring functions serves as a baseline in this work to study human-centered ranking creation based on attribute preferences. A pioneer work for attribute-based ranking creation is *LineUp* [6], an interactive technique designed to create, visualize, and explore rankings of items based on a set of heterogeneous attributes. LineUp enables users to formalize functions that map attribute values to scores, either through a programming interface or through visual interfaces. The visual approach supports the formalization of linear and compound linear (e.g., a roof-function) preferences. However, no interactive visual support is provided for discontinuous functions or categorical attributes. *MyMovieMixer* [25] is an interactive movie recommender system. Users can select filter criteria and apply linear item preferences by using a slider widget. The authors report that users perceived to be more in control of the ranking results by expressing their preferences explicitly. However, MyMovieMixer does not support non-linear preferences. *WeightLifter* [26] is an interactive visualization that allows users to explore the relationship between attribute weights and ranking results, thus increasing the transparency of the ranking model. Users can simultaneously explore up to 10 attributes. However, trade-offs between more than two attributes require attribute grouping to weigh them via sliders, making it difficult for users to precisely express their preferences. Moreover, WeightLifter assumes that attribute values do not require transformation beyond normalization to be considered as attribute scores. *RankViz* [5] is a visualization framework that enables users to compare two rankings and see how each attribute has contributed to the items' ranking positions. Its major downside is that it requires users to have some knowledge about ranking algorithms, thus shifting the focus from a more personalized ranking towards a more interpretable ranking model. *uRank* [7] is an interactive approach for understanding, refining, and reorganizing document items on-the-fly as information needs evolve. Specifically, it enhances predictability through document hint previews, which serve two purposes: allowing users to control the ranking by choosing keywords and supporting understanding by means of a transparent visual representation of scores. To summarize, while promising attribute-based approaches exist, none of the reviewed approaches supports users in expressing all types of desirable preferences [8]. To be able to study commonalities and differences among item-based and attribute-based approaches, we present an extension of RankASco to be used in our proposed experimental study.

2.3. Evaluation of human-centered ranking creation

Approaches for the human-centered creation of rankings are commonly evaluated with usage scenarios [27] and qualitative experiments [28], such as user studies.

Usage scenarios report on how a proposed approach could be used, highlighting the strengths of the approach in solving a specific task. For example, the evaluation of RanKit [4] employed a usage scenario to clearly illustrate the steps from selecting a dataset selection to showing how user feedback is used to improve the ranking. Similarly, Podium [3] leverages a usage scenario to showcase how the approach can be used to identify the most important features of the user's favorite football team.

Qualitative experiments are used to observe and collect feedback on how users interact with an approach in a real-world setting [28]. Item-based approaches have been evaluated by recruiting a number of participants, including both experts and non-experts. The experiments use pre- and post-questionnaires to understand more about how users solve assigned tasks. Attribute-based approaches have been mostly evaluated with expert users, as in the case of WeightLifter [26] and RankViz [5]. One reason may be that the tasks that users aimed to solve have been mostly technical to date. For example, to evaluate RankViz [5], knowledge about ranking algorithms was required to fully understand also the non-visual mechanics.

Hardly any studies have compared item-based approaches with attribute-based approaches. So far, the visualization community does not offer reflections on commonalities and differences of the two types of approaches, and designers of visualization approaches for the creation of ranking algorithms rely on their experiences when it comes to task abstractions, requirement engineering, and iterative visualization and interaction design. A pioneer evaluation approach has been taken by Gratzl et al. with LineUp [6]. The authors conducted a pre-study with just experts using item-based approaches like Microsoft Excel or Tableau, and a post-study with expert and non-expert users using the proposed attribute-based approach. The studies highlighted that novice users were faster in solving the task using LineUp compared to experts using Microsoft Excel or Tableau.

Our experiment goes beyond this scope, as we analyze across-subject item agreement, task completion time, and derive personas as a reflection of our behavioral observations. The usage of *personas* to characterize user behavior is a well-known method in HCI research and practice, such as system design [29], product design [30], and marketing [31]. A persona represents a user group's unique collection of behavior patterns, objectives, and talents as a realistic character to make them more actionable and understandable [32].

3. Scoring functions for attribute-based ranking

The attribute-based ranking approach leverages user preferences regarding attribute characteristics. Expressions automatically have an effect on all items, regardless of the dataset size. To rank items based on attribute preferences, attribute values must be transformed into numerical values that represent the preference *scores* of users. We call this process attribute scoring. For example, users preferring fast cars might favor high HP attribute values, while penalizing low HP attribute values. Ultimately, all attribute-based preference scores can be used and combined to create the overall item ranking.

To perform the mapping from attribute values to preference scores, we build upon Attribute Scoring Functions (ASFs) [8], serving as one of our two baseline workshop publications that we extend in this work. We briefly echo the essentials of ASFs, which are described and discussed in the baseline work in detail. In short, ASFs are mappings of data attributes that:

- *transform* the input values to numerical output scores,
- have a *polarity* for the output score domain, and
- have a *valence* for the output scores.

Data Transformation Each ASF covers the entire input domain of an attribute. This ensures that each attribute value can be mapped to an output score. In addition, any attribute value must be mapped to exactly one output score to ensure the validity of the data transformation and to prevent ambiguity.

Polarity The output score domain of an ASF has a pre-defined range. Similar to normalization, these pre-defined ranges allow for comparable preference scores across attributes. Value ranges

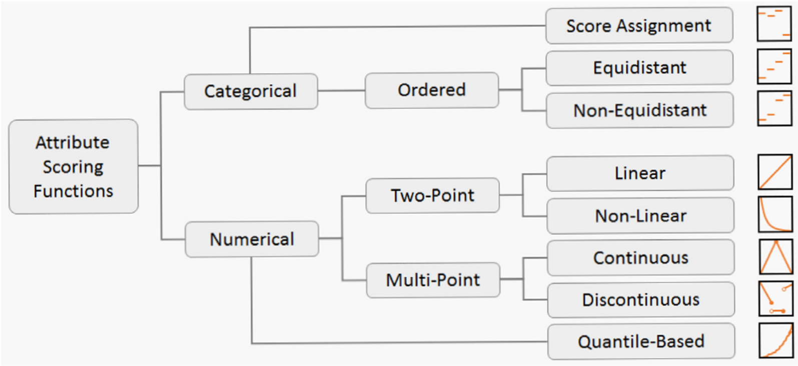


Fig. 1. Taxonomy of eight types of ASFs, used as a functional baseline [8].

can either be uni-polar (e.g., ranging from 0 to +1) or bi-polar (e.g., ranging from −1 to +1). Having a uni-polar range for the output allows users to express how much they like attribute values, while a bi-polar range also allows users to express how much they dislike certain attribute values.

Valence Output scores of ASFs carry valence information, which implies that each output score has semantic meaning. On the one hand, higher scores always represent higher preferences of users compared to lower scores. On the other hand, extreme scores (possibly caused by extreme input values) automatically imply stronger preference values.

We differentiate between categorical and numerical ASFs to explicitly account for the different characteristics of categorical and numerical data attributes. In total, we identified and described eight different types of ASFs in our taxonomy presented in the baseline work, shown in Fig. 1. Three ASFs are applicable to categorical attributes and five are applicable to numerical attributes. For the sake of self-explainability, we briefly re-iterate the eight types.

3.1. Categorical attributes

Categorical ASFs can be used for the transformation of categorical attributes to preference scores. There are three different types of categorical ASFs, which are explained in the following sections: Score Assignment, Equidistant, and Non-Equidistant [8].

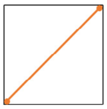
Score assignment. Score Assignment ASFs are the simplest type of categorical ASFs. They work based on absolute preferences, where users directly assign an absolute preference score to each category, in the notion of an explicit quantification [33] of categorical values. This ASF type can be used for assigning exact preference scores to all categories. These scores are absolute, meaning that users can assign preference scores without comparing different categories. A real-world example includes the assignment of scores to different holiday destination cities.

Equidistant. Equidistant ASFs can be used for the assignment of relative preferences to categories. With this ASF type, users can create an order of all categories and assign preference scores to the categories, according to their position in the overall order. The equidistant ASF distributes the score values equally across the value domain. This can be useful if users know about the preferred order of categories, but cannot express how much they prefer a certain category over another. A real-world example includes ordering of different colors for furniture, where users are sure about the order of colors.

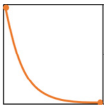
Non-equidistant. Non-Equidistant ASF extend the precision of Equidistant ASFs. They also work based on relative preferences but allow for non-equidistant value score distributions between ordered categories. Especially when several categories of an attribute appear to be similar, non-equidistant ASFs enable users to also assign similar preference scores. For a movie example, the non-equidistant ASF can be used for the ordering of movie genres where users may like very few genres.

3.2. Numerical attributes

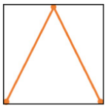
Numerical ASFs can be used for the mapping of continuous numerical attribute values to preference scores using numerical functions. There are five types of numerical ASFs: Two-Point Linear, Two-Point Non-Linear, Multi-Point Continuous, Multi-Point Discontinuous, and Quantile Based [8], all described in the following sections.



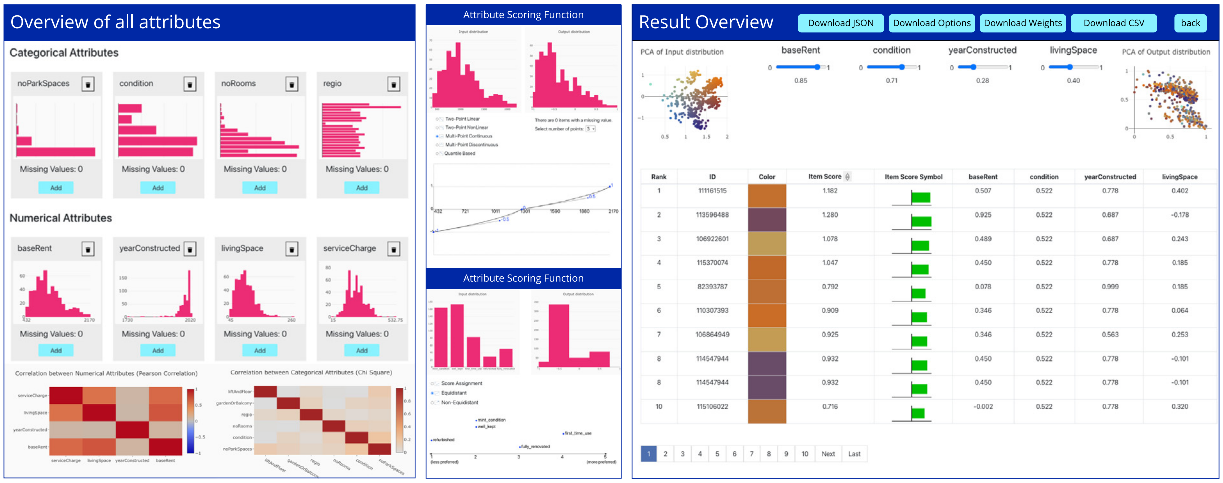
Two-point linear. Two-Point Linear ASFs are a simple type of numerical ASFs and can be used for expressing linear preferences where attribute values at the end of the range (on the top or bottom end) can be favored. This ASF type is suited for non-complex preferences. Examples from the mathematical domain include the min–max or max–min normalization. Real-world examples include the preference for cheapest prices for mobile phone subscriptions.



Two-point non-linear. Two-Point Non-Linear ASFs consist of two points at the start and end of the input range and a line segment in between but, contrary to the Two-Point Linear ASF, can reflect non-linear preferences. This allows users to steer the skewness of the underlying attribute value distribution, enabling users to create ASFs that are similar to, e.g., logarithmic functions or the square root norm. A real-world example is the logarithmic preferences for TV screen sizes, where above a certain point an increase in screen size is only a marginal improvement.



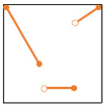
Multi-point continuous. Multi-Point Continuous ASFs expand the design space of ASFs considerably through the addition of additional points within the input value range. Therefore, they allow the creation of more complex and even compound functions. Multi-point Continuous ASFs can reflect sophisticated user preferences that are not monotonically increasing or decreasing, such as preferences for middle values (i.e., roof-like functions) or ramp functions. A real-world example is a preference for middle-priced shoes, since they often have the best price-quality ratio.



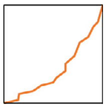
Rank	ID	Color	Item Score	Item Score Symbol	baseRent	condition	yearConstructed	livingSpace
1	111161515	🟧	1.182	🟩	0.507	0.522	0.778	0.402
2	113596488	🟪	1.280	🟩	0.925	0.522	0.687	-0.178
3	106922601	🟨	1.078	🟩	0.489	0.522	0.687	0.243
4	115370074	🟧	1.047	🟩	0.450	0.522	0.778	0.185
5	82393787	🟧	0.792	🟩	0.078	0.522	0.999	0.185
6	110307393	🟧	0.909	🟩	0.346	0.522	0.778	0.064
7	106864949	🟨	0.925	🟩	0.346	0.522	0.563	0.253
8	114547944	🟪	0.932	🟩	0.450	0.522	0.778	-0.101
8	114547944	🟪	0.932	🟩	0.450	0.522	0.778	-0.101
10	115106022	🟧	0.716	🟩	-0.002	0.522	0.778	0.320

(1) (2) (3)

Fig. 2. Overview of the RankASco visual analytics workflow. Users can (1) gain an overview of multiple categorical and numerical attributes and underlying correlations between attributes, (2) create attribute scorings for relevant attributes based on their preferences by using interactive visual interfaces, and (3) configure attribute weights, analyze and refine ranking results, and make informed multi-criteria decisions.



Multi-point discontinuous. Multi-Point Discontinuous ASFs introduce the concept of mathematical discontinuities to the ASF design space. In Multi-Point Discontinuous ASFs not all points must be connected, allowing the creation of functions with gaps in the output domain. A mathematical example of this behavior is a stair function. Real-world examples include the preference for either old-timer cars or the latest car models at the same time (with low preferences for middle-aged cars).



Quantile based. Quantile Based ASFs are different from the Two-Point and Multi-Point ASF types in that they apply statistical quantile normalization to the attribute values. In contrast to value-based functions, the order of values determines the output scores of distribution, similar to the notion of a rolling pin for baking. This ASF type allows users to flatten narrow value distributions, and limit the impact of outliers in the dataset.

4. Abstractions

We briefly characterize the main steps of the workflow when performing attribute-based creations of item rankings, before we describe the rationales that motivated the design of our visual analytics approach. The driving principle was the stringent support for users to express their subjective preferences on attributes, following the goal to create a human-based data analytics solution. The ranking creation workflow is inspired by the work of Wang et al. [34], Kuhlman et al. [35], Gratzl et al. [6], and Cheng et al. [36]. Since our approach is based on user preferences, preferences are the basis of the attribute selection rather than automated selection as in Wang et al. [34]. Overall, we have identified three principal phases in the workflow to create a human-centered ranking, as Fig. 2 illustrates.

1. **Attribute Overview and Selection:** Users should first gain an overview of attributes and select interesting attributes.
2. **Creation of ASFs:** For each selected attribute, users can create an ASF such that their preference for certain attribute values can influence the ranking.
3. **Ranking Analysis:** The ranking is presented to users, enabling the analysis of the validity of the computed ranks.

We articulate seven requirements to visual analytics approaches for the human-centered interactive visual creation of item rankings. These requirements are based on the problem statement, related work on multi-criteria decision-making [5,6,26,36,37], experiences gained through previous work [8], and by echoing human-centered visual analytics principles:

- R1: **Attribute Overview**: Providing an overview of attributes, their value distributions, and dependencies between attributes to support the informed selection of attributes.
- R2: **User Preferences**: Accounting for individual user preferences, creating various ASF types should be supported.
- R3: **Instant Feedback**: Assessing the effect of changed ASFs on underlying data distribution values instantly should be supported for validation and refinement purposes.
- R4: **Straight-Forward ASF Creation**: Opening attribute scoring to a diverse spectrum of users should be supported.
- R5: **Ranking Overview**: Analyzing the ranking results, including influencing scores, should be possible.
- R6: **Attribute Weighting**: Approaches should allow defining and refining the importance of attributes through weights, achieving human-centered rankings.
- R7: **Ranking-Data Comparison**: Assessing how the ranking relates to the underlying item distribution should be possible for users.

5. RankASco – Human-centered attribute-based ranking

We present a visual analytics approach to support users in the interactive creation of human-centered item rankings. RankASco (short for *Rank*ing based on Attribute *Sco*rings) is an attribute-based approach that takes users preferences into account to calculate an item ranking, even for large datasets. We present a refined and extended version of RankASco, as an extension of the original workshop paper publication [9]. An overview of the three main views of RankASco can be seen in Fig. 2, in line with the three main phases of the workflow proposed in Section 4. RankASco is publicly available https://rankasco-ivda.ifi.uzh.ch/, with more implementation details in the supplemental material.

5.1. Phase 1: Attribute overview and selection

The first phase of the interactive workflow for the creation of item rankings consists of the identification of a meaningful

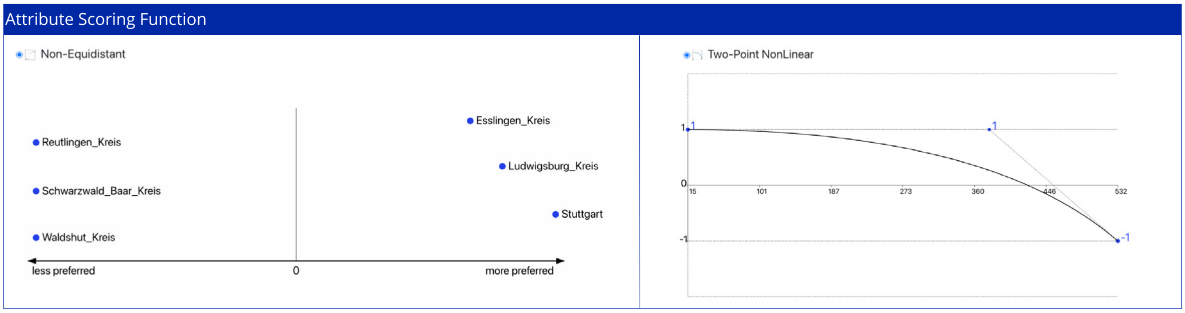


Fig. 3. Two different interfaces for ASF creation. The categorical Non-Equidistant ASF is used to, e.g., express strong preferences for three regions in an apartment-hunting situation (left). A numerical Two-Point Non-Linear ASF shows users preferences for low service charges for the apartment of choice.

set of attributes that are relevant to the users' preferences. The attribute overview and the correlation overview allow users to make an informed selection on a set of attributes (R1). The attribute overview interface in RankASco shows all existing attributes for a given item set, as shown in Fig. 2 (left). For categorical attributes, all categories and their counts are shown in bar charts. For numerical attributes, histograms show the distribution of numerical values. In addition, RankASco also reveals the number of missing values for each attribute. The handling of missing values is crucial to calculate an item score for each item. LineUp [6] handles missing values by calculating the mean or median of an attribute; we use an approach where users can define a score for missing values explicitly. The handling of missing values is different for categorical and numerical attributes, as described in the respective sections. When users select a set of interesting attributes, there likely exist correlations between attributes. To account for this important decision-making criterion, the extended version of RankASco now offers a correlation overview for categorical attributes and for numerical attributes alike, shown in Fig. 2 step 1 (bottom). Categorical correlations are calculated based on the Chi-squared test [38], while the Pearson correlation coefficient [39] is used for numerical correlations.

5.2. Phase 2: Creation of attribute scoring functions

After the identification of a set of relevant attributes, users can create an ASF for each attribute to use for the calculation of the item ranking. The selected types of ASFs are based on the eight different types of ASF that we identified in a baseline work [8]. RankASco supports this stage by providing eight different interactive visual interfaces for the creation of the eight different types of ASFs, which is the core of the baseline publication [9]. With the eight visual interfaces, a broad spectrum of mental models of users can be addressed (R2): Some ASF-creation interfaces are simple and straightforward, while other variants are more complex and highly customizable. To guide users in the selection and the creation of an ASF, visual fingerprints explain the functional behavior of the ASFs and respective interfaces. This helps users find the best ASF type for their preferences and the underlying attribute data.

The design of all eight ASF interfaces follows the same principles: Input values (the attribute values) are shown on top left in the ASF creation view, output values (the output scores) are shown on top right, next to the input values as can be seen in Fig. 4 (left). This eases the comparison between the characteristics of the input and output value distribution, and thus the effects of the ASF on the data attribute. The actual ASF-creation interface is always shown below the two distribution charts and differs for all eight types of ASFs. The iterative process particularly focused on the design of interfaces that are easy to use (R4). Direct manipulation and linking of views update the output value distribution in real-time whenever the ASF is modified (R3). Design and implementation details of the eight interfaces for ASF creation are as follows.

5.2.1. Categorical attributes

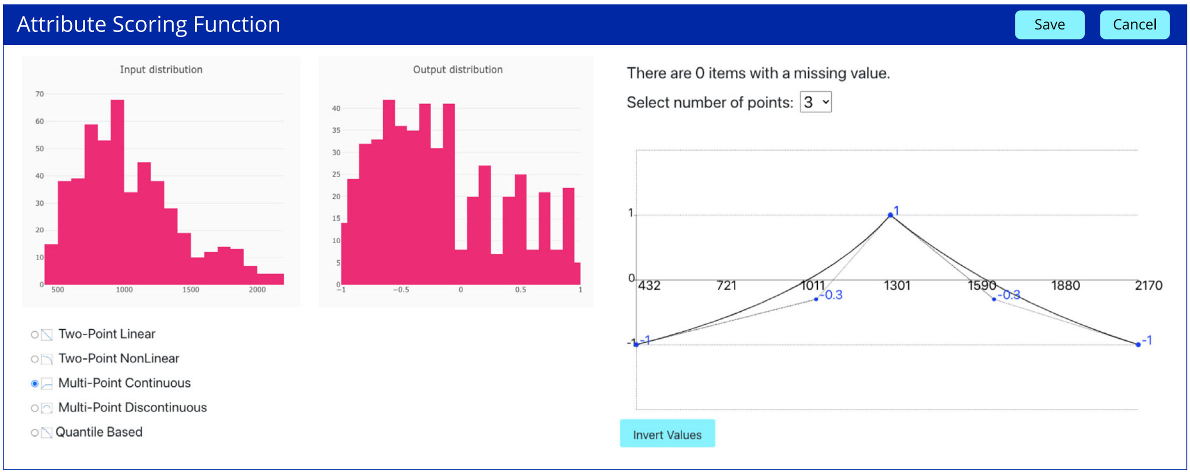
Three interfaces allow users to create ASFs with preference scores for categorical attributes. All three interfaces share the same strategy to support missing value treatment: missing values of categorical attributes always form an additional category that can be considered by users for scoring purposes.

The *Score Assignment* ASF is based on absolute preferences. It allows users to assign a numerical preference score to each category. In RankASco, this ASF type is represented through numerical input fields (one for each category in a categorical attribute) where users can directly assign preference scores between -1 and $+1$. To ease the usage, users can also start with a pre-defined neutral value for all categories and assign preference scores for a subset of categories only. This feature overcomes the need for setting a score for every category, even if irrelevant. This is especially efficient for categorical attributes of high cardinality.

The *Equidistant* and *Non-Equidistant* ASF types are based on relative preferences. The interface of both ASF types are two-dimensional, where categories are shown along the y-axis and preference scores are shown along the x-axis. Users can adjust the position of each category by horizontal dragging interaction, from the left (less preferred) to the right (more preferred). The difference between the two ASFs is the placement strategy of categories along the x-axis: for the *Equidistant* ASF, categories are positioned along discrete equidistant positions, to guarantee equal spacing between categories. In contrast, with the *Non-Equidistant* ASF, users have the ability to position categories continuously along the x-axis, allowing for non-equal spacing between categories. Fig. 3 (left) shows a Non-Equidistant ASF that represents users preferences for certain regions. A detailed example for the Score Assignment and Equidistant ASFs can be found in the supplemental material.

5.2.2. Numerical attributes

The interfaces for the four value-based numerical ASF types (all numerical ASFs except *Quantile based*) use a two-dimensional coordinate system. Attribute values are shown on the x-axis and preference scores are shown on the y-axis. This design choice is based on mathematical functions $f(x) = y$ and how they are visualized in 2D. The interfaces for each ASF type initialize a default function that can be adjusted with draggable points, e.g., to steer the slope and curvature of the line segments between points. The user-created function determines how input values are transformed into preference scores. An example of each of the four numerical ASFs can be found in the supplemental material, including an enlarged figure.



There are 0 items with a missing value.

Select number of points: 3

○ Two-Point Linear
○ Two-Point NonLinear
● Multi-Point Continuous
○ Multi-Point Discontinuous
○ Quantile Based

Fig. 4. One of five visual interfaces for the creation of numerical ASFs. Here, a Multi-Point Continuous function is used to represent the user's preferences for base rent prices around €1300 (drag-and-drop interface on the right). On the left, two histograms show the distribution of input values and output scores. This instant feedback also helps to achieve balanced scores, compared to the left-skewed input values.

The *Two-Point Linear* ASF consists of one linear line segment spanning across the entire input value domain. Two points at the very left and very right of the *x*-axis can be vertically adjusted, to change the slope of the ASF. An example of a *Two-Point Linear* ASF can be found in the supplemental material.

The *Two-Point Non-Linear* ASF type expands this concept by allowing for a non-linear line segment between the two points. The curvature of this line segment can also be steered through an additional point, a so-called control point, based on the mathematical concept of Bézier curves [40]. Fig. 3 (right) shows a Two-Point Non-Linear ASF that shows a non-linear (logarithmic) preference for service charge values.

Multi-Point ASF types have more than two points and also more line segments, respectively. Thus, they allow more flexibility in function design. The mode of operation is the same as for the *Two-Point* ASFs: The line segments and their curvature can be steered through draggable points, as shown in Fig. 4. For the *Multi-Point Continuous* ASF, all lines are always connected to each other, resulting in a continuous mathematical function. *Multi-Point Discontinuous* ASFs, on the other hand, introduce mathematical discontinuities between line segments to create gaps in the output domain.

The *Quantile Based* ASF works based on statistical quantile normalization, which is applied to the order of the attribute values instead of the actual values. One insight we had in the design process was to allow steering the degree to which an input value distribution shall be subject to quantile normalization. The interface now offers a slider that lets users steer the degree of quantile normalization that is applied to the data, ranging from 0% (no quantile normalization at all) to 100% (full quantile normalization applied).

5.3. Phase 3: Ranking analysis

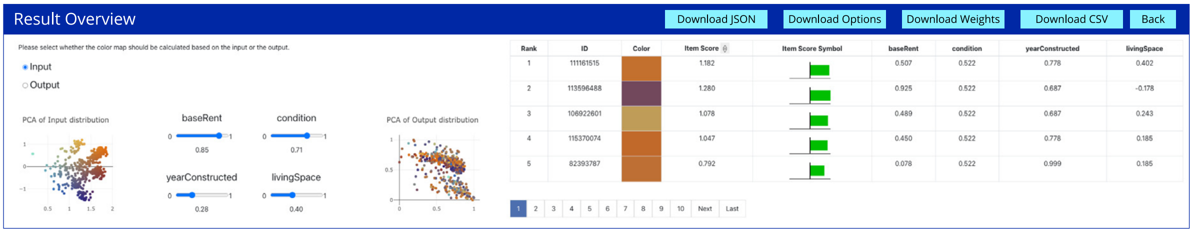
The final phase of the ranking creation workflow is the analysis, validation, and possible refinement of the created ranking. Based on the set of created ASFs and a user-steerable weight for each attribute, an overall item score is calculated for each item. This score is a weighted sum of all the attribute preference scores multiplied by their attribute weight (more details are given below). The final item ranking then results from ordering all item scores in decreasing order.

The design of the ranking interface is inspired by list-based item visualizations, typically utilized in interfaces for search results [41–43], and the output of recommender systems [2,44,45]. The ranking result is split into multiple pages, which allows users to either only look into the top items or, if interested, also check items ranked in the middle by using the pagination. This visualization allows to only print the top items first and, if requested, load additional items to handle large item sets better. To account for details that help explain item ranks, we extend the list-based idea to a tabular layout, as shown in Fig. 5 (right). This table contains all items (rows) as well as columns for the attribute scores involved in the human-centered ranking process (R5). Users can steer the weights for all attributes with sliders, as shown in Fig. 5 (left). These adjustable weight sliders allow users to assign preferences of importance to the different attributes and are initially set to 0.5. Modifying one of the attribute weights results in a re-calculation of all item scores in real-time and an update of the item ranking (R6).

The weighted score for each item is supported with a visual cue that eases the comparison of different items, as shown in Fig. 5. Additional scatter plots support the comparison of the input and output data characteristics (R7): The input data (Fig. 5 left) consists of a dimensionality-reduced version of the one-hot encoded original dataset. The output data (Fig. 5 right) consists of a dimensionality-reduced version of all attribute scores for each item. Every item also has a unique ID and a color that is determined in a similarity-preserving way. Colors are assigned based on a 2D color map [46] on either the input or output distribution of all items, as shown in the scatter plots. Linking items between one of the two scatter plots and the ranking result facilitates the comparison of top-ranked items and their distribution originating from the input or output data distributions. To finish the interactive ranking creation workflow, users can export the ranking (CSV or JSON format) and use it for downstream analyses, as shown in Fig. 5 (right).

6. Usage scenarios: Apartment-hunting

We introduce two usage scenarios for multi-criteria ranking problems with a dataset about apartments in the south-west of Germany, publicly available at Kaggle [47]. We will be talking about the Fischer family: Hugo Fischer, husband of Barbara Fischer and father of two girls. Hugo is a fictive non-expert who is looking for a new apartment for his family. Overall, the Fischer family has ten preferences on apartments with different priority, pertaining to ten different attributes. This scenario will recur in our experimental study; details on exact preferences of the Fischer family are described in the supplemental material.



Please select whether the color map should be calculated based on the input or the output.

● Input
○ Output

baseRent 0.85
condition 0.71
yearConstructed 0.28
livingSpace 0.40

Rank	ID	Color	Item Score	Item Score Symbol	baseRent	condition	yearConstructed	livingSpace
1	111161515	🟧	1.182	🟩	0.507	0.522	0.778	0.402
2	113596488	🟪	1.280	🟩	0.925	0.522	0.687	-0.178
3	106922601	🟨	1.078	🟩	0.489	0.522	0.687	0.243
4	115370074	🟧	1.047	🟩	0.450	0.522	0.778	0.185
5	82393787	🟧	0.792	🟩	0.078	0.522	0.999	0.185

1 2 3 4 5 6 7 8 9 10 Next Last

Fig. 5. Ranking result overview with color-coding across views. The interface on the left shows the data spaces of the input and the output scores (scatter plots) and allows steering attribute weights. The ranking result interface on the right shows details on attribute scores, with similarity-preserving colors.

6.1. Apartment-hunting with the general purpose tool

Hugo takes an item-based approach, using a spreadsheet tool such as Microsoft Excel, to organize the apartment items in a preferred way. He decides for the spreadsheet tool because he owns a software license for the tool anyway and has been using Excel occasionally during the last years. The tabular format gives him control over the items (rows), while always having the lookup of attributes (columns).

First, he starts with gaining an overview of the dataset. Vertical scrolling helps him traverse all items, and he realizes that the number of available apartment items is large. Horizontal scrolling lets Hugo identify preferred attributes and delete irrelevant attributes to reduce task complexity. Next, he uses a filter to reduce the dataset size: for the *region*, he excludes regions other than “Stuttgart”, “Ludwigsburg Kreis”, and “Esslingen Kreis”, which reduces the dataset size by roughly 80%. Next, Hugo uses a filter in the notion of a dynamic query [48] to remove *base rent* values below €1000 and above €1600, as the allocated budget of the family is about €1300. As a third operation, Hugo selects the *rooms* column and filters out *room sizes* outside the values 3.5, 4, 4.5, and 5. He then sorts the *service cost* charges from least to most, as low additional costs are important for the family. After these operations, Hugo notices that only 26 flats remain for decision-making. Possibly, Hugo’s filter criteria have been too stringent for one or the other attribute, such that valuable items may have disappeared; he therefore relaxes the filter criteria a bit. Also, Hugo is not yet happy with the ranking order of items. Hugo decides to sort by the *base rent* attribute due to its high importance and discovers that the attribute has a bipolar nature, meaning that the best apartments around €1300 are not at the top. With some scripting effort, he fixes the problem and, with some additional scripting, manages to sort items by more than one attribute at the same time. Finally, to take all ten preferences on apartments into account, he starts with changing the order of items manually to arrive at a final ranking. Hugo shows the result to Barbara, and together they determine which of the top flats they want to visit as a family.

6.2. Apartment-hunting with RankASco

We demonstrate the usefulness of RankASco for the multi-criteria ranking problem of the Fischer family. We will accompany Hugo’s workflow until he has created ASFs for the four preferences of highest priority.

Hugo starts using RankASco by analyzing all attributes and attribute value distributions shown in Fig. 2 (left). He is particularly interested in the *region*, *base rent*, *number of rooms*, and *service charge*; so he starts with the *region* attribute. He creates the Non-Equidistant ASF shown in Fig. 3 (left) representing his strong preference for the three regions “Stuttgart”, “Ludwigsburg Kreis”, and “Esslingen Kreis” (in that order). Next, Hugo picks the *base rent* attribute and creates the Multi-Point Continuous ASF, depicted in Fig. 4 (right). The roof-like function punishes apartments that are too cheap. Beginning with €1000, apartments turn positive, with a maximum at €1300. Even larger prices for rent turn into negative scores at €1600. Then, Hugo defines his preference on 4-room flats (with 3 to 5 *rooms* also deemed acceptable), using a Score Assignment ASF. Next, Hugo chooses the *service charge* attribute with a preference for service charges as low as possible, represented with a Non-Linear ASF shown in Fig. 3 (right). The non-linear nature of the function returns positive scores for many of the low values of service charge, but decreases steeply for very high values. This is an example of how Hugo can exploit the bipolar support for scores given with RankASco (polarity characteristics).

After creating the four ASFs, Hugo proceeds to the ranking overview and starts with refining the attribute weights per attribute, as shown in Fig. 5 (left). Given his preference scores and weights per attribute, RankASco automatically provides the resulting item ranking (Fig. 5 (right)). From here, Hugo’s remaining process is three-fold. First, Hugo can refine the four created ASFs, if the analysis of the ranking result reveals aspects that can be improved. Second, he uses RankASco’s export functionality to show the preliminary list of top candidates to his wife Barbara. Third, he continues with adding the missing six preferences to arrive at the final ranking according to the Fischer’s preferences, as a start for the informed visit of quasi-optimal apartments.

7. User study

With the proposal of a visual analytics approach for the human-based creation of item rankings, we widen the bandwidth of existing approaches in a still loosely populated design space. Interesting questions emerge regarding the evaluation of RankASco, but also with respect to human factors involved in the ranking creation realm. For that purpose, we conducted an experimental study with two distinct parts: the observation and analysis of user performance, and the observation of and reflection on user behavior. The first main goal of our experiment was to compare RankASco with a general purpose tool, similar to the user study of LineUp [6] with Excel and Tableau. We crosscut this performance analysis with our second goal: to observe and identify user behaviors among study participants to ultimately derive personas. In the study, data collection included quantitative and qualitative data by taking participants’ task completion time, determining across-subject item agreement in the top 20 ranking results, recording behavioral observations, and conducting informal interviews. We first describe the research questions and the experiment design, before we provide details on the results of the two study parts in Section 8.

7.1. Research questions

The two main goals can be broken down into four research questions as follows:

RQ_1: Can a stringently attribute-based ranking approach compete with the general purpose tool in terms of efficiency?

RQ_2: Does the number of items have an impact on the performance of the attribute-based ranking creation in comparison to the general purpose tool?
RQ_3: How do users behave in the three different phases of the ranking workflow?
RQ_4: Is it possible to derive personas from observed user behavior in both RankASco and the general purpose tool?

RQ_1 and RQ_2 are related to the quantitative assessment of user performance (Part 1), while RQ_3 and RQ_4 are related to behavioral observations of users (Part 2).

7.2. Experiment factor: RankASco and general purpose tool

We use RankASco as a representative of a multi-criteria attribute-based ranking tool for the interactive creation of item rankings. The decision for RankASco is based on its completeness in the support of categorical and numerical attributes through interactive visual interfaces to support eight types of ASFs and its stringent design for large user groups, including non-experts. In contrast to, e.g., LineUp [6], RankASco is the only attribute-based ranking approach that entirely works without the need for coding. It also supports numerical and categorical attributes. To allow for a fair comparison between RankASco and the general purpose tool, the correlation plots in RankASco were disabled for the user study.

We used a general purpose tool as a representative of the different options with which users can create item rankings in their everyday live. Those include filtering by many and sorting by one attribute. We aimed for an approach for the creation of item rankings that should neither require expert knowledge nor programming skills. An overview of items and attributes should be provided, such that users can make informed decisions on the ordering of items. Finally, users should be able to express preferences through direct manipulation. Similar to Gratzl et al. [6], we used Excel due to its popularity for the targeted user population.

7.3. Experiment factor: Dataset size

The underlying dataset forms the basis for a second experiment factor: the dataset size. We utilized the Kaggle "Apartment Rental Offers in Germany" [47] dataset for the experiment. Overall, the dataset contains 268,850 apartment listings (items) with a total of 49 attributes. After the exclusion of binary, range, redundant, ambiguous, and task-irrelevant attributes, we chose six categorical and four numerical attributes for this study. In an upstream process, we made data quality checks and eliminated items that contained null values, missing attributes, or implausible values (cf. supplemental material).

To study user performance with respect to the dataset size, we control the number of items as one experiment factor. From the remaining items of sufficient quality, we randomly selected 500 items. To arrive at different experiment conditions, we used these items to create three subsets: *500 items*, *300 items*, and *100 items*.

7.4. Participant description

We recruited 24 participants (14 female) at the university, aged between 22 and 31 ($M = 26$, $SD = 3.09$). A prerequisite for the experiment was a basic command of Excel and the ability to understand and speak the offered experiment languages (EN & DE). Human subjects research approval from the faculty's ethics board was obtained prior to the study. Participants who completed the study received a gift card worth $/€30 as compensation. Prior to the study, we asked participants about their knowledge in Excel ($M = 3.29$, $SD = 0.94$), data science ($M = 3.29$, $SD = 0.84$), and multivariate data analysis ($M = 2.79$, $SD = 1.00$), using a 5-point Likert scale (high signifies very good knowledge). Additionally, we asked whether the participants had prior experience creating rankings for decision-making problems (29% *yes*) or have already solved a ranking problem programmatically (25% *yes*).

7.5. Task description

The task for all participants was to create a ranking for a given set of items and the tool at hand. To facilitate the comparability of results, we controlled the preferences that participants would have to follow in the experiment, i.e., we introduce the truth of the ranking scenario upfront. We designed a narrative evaluation [49], where participants assumed the role of a real estate agent who is aiming at identifying the top 20 apartment items, based on the preferences of the Fischer family, the clients of the real estate agent (cf. the Usage Scenario in Section 6 and the supplemental material). We designed the preferences of the Fischer family based on two main goals. First, the preferences should include a healthy mix of attribute characteristics, with preferences ranging from simple (e.g., "the higher, the better") to complex (e.g., mathematical discontinuities like "preferred if apartment is built before 1900 or as new as possible"). Second, the preferences should all be plausible for a family with two kids. Overall, ten preferences needed to be considered, each for a different attribute. We created a tabular description of the preference attributes, sorted by their importance.

7.6. Dependent and independent variables

Independent variables are the *type of ranking approach* (using the general purpose tool or the RankASco) and the *dataset size* (100, 300, or 500 items). The crosscut of the two variables leads to six experiment conditions. Every participant was assigned to only one dataset size level, i.e., eight participants were tasked with 100 items, etc. In contrast, every participant was asked to perform the ranking task on both types of ranking approaches to maximize the comparability across approaches. To avoid the learning effects and effects of fatigue, we randomized the tool to start with between participants for all three dataset sizes. Dependent variables are the *task completion time* and the *across-subject item agreement* of the top 20 ranking results.

7.7. Study procedure

We carried out a pilot study in advance to make sure that task and study design were understandable, robust, and feasible. The study procedure included four steps: (1) introduction, (2) training, (3) ranking creation, and (4) questionnaire. We introduced participants to their task by providing a narrative and dataset description, to ease the lookup of preferences of the Fischer family whenever needed (cf. supplemental material). Then, we conducted an introduction session to the approaches used so that participants could always familiarize themselves. Participants were trained by walking them through the interfaces of the tools. Using the movies dataset we explained each ASF with a usage example.

In the core of the study, participants solved the ranking task with the two approaches. In parallel, we conducted an observational study to also assess the user behavior. We measured the participants' *task completion time* without prior announcement, to avoid time pressure on the participants' side. To assess *across-subject item agreement*, we collected the final top 20 items after participants completed the ranking task. Finally, we conducted a qualitative interview utilizing a 5-point Likert scale rating to

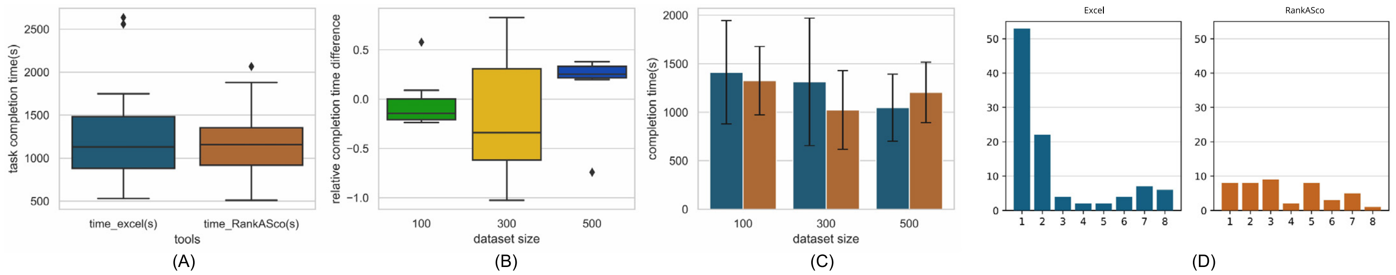


Fig. 6. (A) Comparison of task completion time of RankASco versus general purpose tool. (B) Relative difference of task completion time, depending on the number of items. Values > 0 indicate that using RankASco is faster, whereas values < 0 indicate that using the general purpose tool (Excel) is preferable. (C) Task completion time for the subsets of 100, 300, and 500 items. (D) The heterogeneity of the top 20 ranked items across participants withing the 100 items dataset (for the 300, 500 items see the supplemental materials). Here, 1 indicates that no other participant agreed with that item being in the top 20 ranks (high heterogeneity), whereas 8 indicates that a particular item was in the top 20 items of all participants (low heterogeneity).

assess and compare participants' perceived confidence in their ranking results (cf. supplemental material). Interview questions also included the users' experience with the two approaches and personal preferences on approach usage (cf. supplemental material).

7.8. Data analysis

Part 1: Performance analysis (RQ_1, RQ_2). We analyzed the performance measures for (a) the comparison of the two item ranking approaches (RQ_1), (b) the comparison of the three dataset sizes (RQ_2), and (c) the cross-cut of both experiment factors (2x3 conditions) (RQ_2). To perform this data analysis strategy, we used a two-fold approach. First, we used visual representations for (a-c) to assess effects visually (see Fig. 6). Second, we applied statistical tests to identify considerable or even significant differences between the conditions with respect to the dependent variables. The test portfolio included a paired two-sample t-test [50] and Wilcoxon Rank-Sum Test [51–53] to compare item ranking approaches regarding both measures (a). We also performed a one-tailed ANOVA test [54] to assess differences between the three dataset sizes (b). Input for the test was the relative differences of task completion times for the two item ranking approaches.

Part 2: Assessment of user behavior (RQ_3, RQ_4). To assess participant behavior, two authors coded the study observation notes and extracted behavioral variables [55–57]. Coding conflicts were resolved by a third author not involved in the coding process. Finally, behavioral variables were reviewed by an external researcher not involved in their creation. Overall, we distinguished between general behaviors observed in both approaches, behaviors observed only in the item-based ranking approach, and unique behaviors of the attribute-based approach. Re-iterating over the study observation notes, one author further assigned participants a score between 1 (not present) and 5 (very pronounced) for each behavioral variable derived. Ultimately, we used the observational data, participant knowledge assessments, and behavioral variables with their manifestations for the identification of personas (RQ_4). For the analysis of interactions between the two approaches (item-based and attribute-based), behavioral variables, and personas, we created two heatmaps shown in Fig. 7.

8. Results of the user study

8.1. Part 1: Performance analysis (RQ_1 , RQ_2)

Fig. 6(A) shows task completion time with RankASco and with the general purpose tool (RQ_1). Clearly, using the general purpose tool resulted in greater variability and outliers for participants. Conversely, independent of the size of the dataset, there is less variability regarding completion times when using RankASco. However, there was no statistically significant difference in the average task completion time ($t(23) = -0.588$, $p = 0.563$).

Fig. 6(B) includes box plots of the relative completion time difference for RankASco versus the general purpose tool, across three dataset sizes (RQ_2).

Values > 0 indicate that a participant was faster when using RankASco versus the general purpose tool and vice versa for values < 0.

Looking at the visualization depicted in Fig. 6(B), three findings stand out. First, participants were on average faster using the general purpose tool when the dataset contained 500 items (median > 0). Second, participants were faster using RankASco when using the 100 items set (median < 0). The third finding is that for 300 items dataset the task completion time was most diverse for the general purpose tool. In summary, our assumption that RankASco as an attribute-based approach performs better for larger datasets was not observed. We believe that different types of user behaviors had a stronger effect on the task completion time than the dataset size.

Fig. 6(C) reveals that it took participants longer using a smaller item set (RQ_2). One explanation of this finding could be that for only 100 items, several participants did take the time to traverse and interpret the entire item collection, in contrast to larger item sets. Another possible explanation for this unexpected finding is that randomly selecting a subset from the 500-item dataset reduced the number of suitable apartments substantially. As a result, few to no items remained after implementing all the preferences set by the Fischer family in Excel. Confronting this problem could have reinforced the expression of user behavior, as discussed in the next section. Some participants adopted a very pragmatic approach to the issue: *"After realizing that there is no optimal solution that can be found with the filters, I only considered the two most important preferences and disregarded all the others".* - P12. Other participants went over each item one at a time, in an effort to see how they could best balance these subpar results on a per-item level. We assume that this can be considered a turning point that leads to the higher variance regarding participants' time in Excel, as shown in Fig. 6(C). A one-way ANOVA revealed that there was a statistically significant difference in the percent difference of completion time between at least two dataset sizes ($F(2, 24) = 6.43$, p< 0.01). Tukey's HSD Test for multiple comparisons showed that the mean value of the percent difference was significantly different between the 100 and 300 item datasets (p< 0.01, $95\%\ C.I. = [0.113,\ 0.680]$), and notably different between the 300 and 500 item datasets ($p = 0.074$, $95\%\ C.I. = [-0.545, 0.022]$). This underlines the special nature of the 300-item dataset.

Fig. 6(D) shows the across-subject item agreement regarding the top-20 ranks of participants assigned to the 100 item dataset (N = 8) for the two ranking approaches (RQ_1). The results of the remaining participants (N = 16) for the 300 and 500 item datasets can be found in the supplemental materials. The number on the x-axis signifies how often an item-was picked by participants, where 8 indicates that an item was picked by all participants and one that an item was only picked by one participant. Fig. 6(D) clearly shows that the attribute-based approach had greater across-subject item agreement with the 100 item dataset: participants picked 100 and 44 different items for their top 20 items using the item-based and attribute-based approach, respectively, 44 items in common between approaches. A Wilcoxon Ranked-Sum test revealed that the distribution of across-subject item agreement between the ranking results of RankASco versus the general purpose tool is significantly different for the 100 items dataset ($U = 3.6$, $n_1 = 44$, $n_2 = 100$, $p < 0.01$ two-tailed). Differences regarding the across-subject item agreement in the top 20 ranks and item count distributions were not notable for the 300 and 500 item sets (results in the supplemental material).

Based on our performance analysis, the selection of an appropriate ranking approach should be influenced by the characteristics of the dataset, including its size and attribute composition. Our findings indicate that RankASco produces superior results in terms of completion time variability and across-subject item agreement. The smaller completion time variability suggests that RankASco may be particularly well-suited for use with a heterogeneous user pool. We discovered that the difficulty of the ranking task is influenced not only by the complexity of the user preferences but also by the items present in the dataset at hand. For multiple criteria to be considered or item sets which do not match preferences nicely, users may have to start compromising their preferences, which adds to the item ranking challenge. In these cases, using an attribute-based approach like RankASco may be preferable, as users can refrain from tedious and complex item-level comparisons.

In summary, we found partial evidence to confirm RQ_1 and RQ_2. However, we also learned that the behavior of individual users may have a much bigger effect on ranking effectiveness and efficiency as assumed. Informed by this finding, we next present the assessment of user behavior.

8.2. Part 2: Assessment of user behavior (RQ_3, RQ_4)

We structure the assessment into low-level behavior of users and high-level personas, as an abstraction of user behaviors.

8.2.1. Low-level behavioral variables (RQ_3)

We observed many behavioral patterns in how participants addressed the ranking-creation task, which we distilled into behavioral variables shown in Table 1. Most participants repeatedly used the task description in between the steps to create the item ranking. While most participants marked important statements in the narrative, some participants spent a lot of time manually weighting the different attributes in the task description. Only after pre-processing was completed, the participants' timing started. The full description of the observed behaviors and interview results can be found in the supplemental material. A concise version is depicted in Table 1.

8.2.2. High-level personas (RQ_4)

We report on the discovery of three personas, based on observations of participants in the study. These personas are the result of coding the observational data and studying the behavioral variables [32] and their manifestations, which are depicted in Table 1. We then summarized similar manifestations of the behavioral variables, which resulted in three personas: (1) Peter, the perfectionist (N = 10), (2) Eva, the explorer (N = 7), and (3) Pippa, the pragmatist (N = 7). In the following, we present each persona in detail.

Table 1
Short description of the behavioral variables (extended version, cf. supplemental material), separated by with which approach they were observed: using RankASco (R) or the general purpose tool (G).

	Behavioral variable	Description
G	Comparing Item Details	Describes the degree to which participants took item details into account.
	Neglecting Preferences	Describes the number of preferences ignored by participants while creating the ranking.
	Query complexity	Describes the degree to which filtering and sorting operations were applied.
	Softening Preference Specifications	Describes the behavior to allow flexibility for some attribute preferences.
	Top-Rank Determination	Describes the speed at which participants were willing to decide on a winning item.
	Undo	Describes the willingness of participants to refrain from actions taken and re-iterate.
	Grouping Items	Describes the strategy to create and elaborate on subsets in the items.
R	ASF Interface Exploration	Describes the extent to which participants explored the design space of the given approach.
	ASF Creation	Describes how precise participants tried to match the preferences with the ASFs.
	ASF Fine-Tuning	Describes how much fine-tuning participants performed when refining an ASF.
	ASF Heterogeneity	Describes the heterogeneity of ASF types the participants used to create the final ranking.
	Neglecting Preferences	Describes the degree to which participants ignored preferences given through the task.
	ASF Resetting	Describes the frequency of participants resetting an ASF to a previous state.
	Considering Input/Output Effects	Describes whether participants used the input/output distribution charts in the ASF creation process.

Peter is a *perfectionist*. He strives to meet all standards of his objectives with the utmost accuracy. He often and thoroughly checks the original specifications of a task throughout the process. Also, he tends to conduct micromanagement along the way, which is why time management can be problematic for Peter. Using RankASco, his goal is to create the most precise and detailed ASFs to represent the preferences of the Fischer family as accurately as possible. Peter might not use all capabilities of a tool, but rather optimizes the output with the capabilities that he is aware of, thus going towards a local optimum. He may become discouraged if the interactive options offered, e.g., for the ASF creation do not satisfy his need for perfection. For Peter, using Excel for item ranking is not easy, as the means to express general ranking preferences formally are missing. Also, the number of pairwise item comparisons needed with the general purpose tool is a challenge, especially for increasing dataset sizes. Overall, we observed the tendency among perfectionists to prefer RankASco over Excel.

Eva is an *explorer*. Her goal is to try out and experiment with all capabilities of a given tool before concentrating on resolving

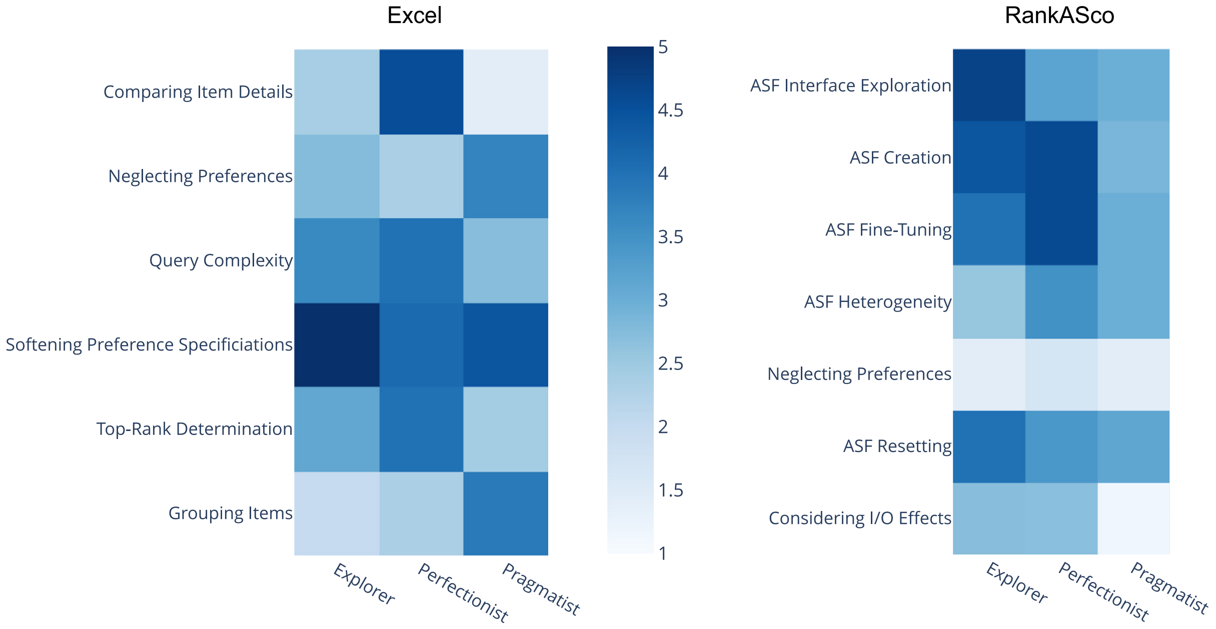


Fig. 7. Behavioral variable manifestations of participants using RankASco or the general purpose tool (Excel), grouped by the three personas.

the task at hand. Her exploratory nature helps Eva gain in-depth knowledge of the approach's design space, helping her assess which functionality is best suited to address her goal. As a downside, Eva may lose sight of her goal. Using RankASco as an example, Eva first experiments with all different types of ASF-creation interfaces before creating actionable attribute preferences. Also, Eva will reset, refine, or even undo intermediate results during the process to fully investigate and finally exploit the capabilities of the visual interface. We observed this in both RankASco and Excel. Fine-tuning and achieving the most meaningful ASF is not her highest priority. Using Excel, Eva heavily applies filtering and sorting functionality, and she accepts that some actions may turn out not useful and need to be reverted. If this notion turns too much into a try-and error manner, her approach may be time-consuming. As a result, in time-critical situations, a less complex application may be preferable to avoid distraction and to help explorers streamline their actions. Overall, we observed that explorers can get lost in the functionality provided by the ranking tool at hand, but they can work effectively with both RankASco and Excel.

Pippa is a *pragmatist*. She wants to get things done efficiently. When problems arise, she is willing to compromise on preferences and accept lower-quality task completion. Her approach to problems is straightforward and linear, i.e., rather less looking to the left and right. Pippa applies a clear and rigorous prioritization of preferences, while possibly neglecting preferences of lower priority. Pippa's approach hardly involves resetting actions or decisions made, as she puts less emphasis on fine-tuning, refinement, and reflection. In RankASco, she initially selects the ASF that she believes to be the most practical, e.g., Two-Point Linear, and tends to use this functionality repeatedly, even if some preferences of the Fischer family would require more appropriate ASF types. When using Excel, Pippa is among the fastest to complete the task, regardless of the dataset size. The reason is simple: Pippa does not systematically inspect all items given, but is fine with seeing some promising items at the top. This has a strong positive effect on task completion time but a negative effect on task performance. In Excel, Pippa is also one of the first to discard preferences if they are contradicting, overly complicated, or fail to yield the desired outcomes. This type of complexity is what Pippa would like to avoid, due to her practical and pragmatic nature. Overall, we observed that using the general purpose tool was more suitable for Pippa, as she was less keen on considering multiple attributes in parallel for decision-making.

The user groups and associated personas show that using RankASco is most appropriate for meticulous and perfectionist users. For more pragmatic users, the general purpose tool is more suitable, as the per-item operations were preferred over multiple attribute-based actions needed. Finally, using the general purpose tool tends to be faster for exploratory users, while using RankASco performs better in terms of confidence in the rankings produced and approach usability, highlighting the difference between speed and perceived success.

9. Discussion and future work

Personas. We have identified three personas based on the observation of participants across approaches and dataset sizes. As we derived the personas as a result from the higher-level analysis of user behavior *after* the study, we did not have the chance to systematically analyze personas *during* the study, e.g., with respect to the usage of the eight types of ASFs, which could be insightful. Looking forward, it would be interesting to design and develop future ranking approaches with the awareness for personas. Interesting decisions include determining if every persona needs its own design, or if future approaches manage to incorporate and support all three personas.

Experiment: Selection of approaches. We have decided for RankASco as the representative of a multi-criteria attribute-based ranking tool and Excel as a representative of a well-known general purpose tool used in everyday-live situations. Although this was well thought through and led to interesting findings, one difference between these approaches is the novelty of RankASco. In contrast to Excel, the learning curve of participants for RankASco needed to include both tool familiarization and task adoption. Beyond Excel and on the long run, different tools with different support for item-based and attribute-based interactions may exist, which would be worth studying.

Experiment: Study design. We designed the experiment in a way that every participant was asked to use both approaches (randomized) for one pre-determined dataset size (100, 300, or 500 items), as the comparison between the general purpose tool and RankASco was key. As an alternative, the randomized assignment of users to dataset sizes 100, 300, and 500 items may have revealed stronger results on the assessment and usefulness of the two approaches with respect to dataset size. The assumption that using RankASco would scale better for large datasets was not found, possibly due to other influencing aspects that require a clearer characterization, such as the pragmatism persona or the sampling method for 100 items. Pertaining to the assumption, future work includes determining the break-even point where stringently attribute-based approaches outperform other ranking approaches, which are less agnostic to the item count.

No quantitative assessment of accuracy. The accuracy of users when working with different approaches and dataset sizes was difficult to assess quantitatively. The reason for this is the lack of ground truth information for the preference-based item ranking case, which does not allow for a quantitative performance evaluation in that regard. The assessment of accuracy, relevance, precision, or similar measures known in machine learning, information retrieval, and similar, is a subject of future work. We identify the lack of clearly defined and formalized ground truth scenarios for ranking creation and we are working on methodologies to address this.

When am I finished? The subjective ranking-creation task based on preferences of the Fischer family is one out of many possible ranking creation goals. We have observed an interesting pattern across participants. This multi-truth situation is difficult to validate and the procedural perspective revealed challenges for many participants: when is a ranking-creation task finished? In general, we believe that the class of preference-based creation/modeling/learning tasks may benefit from process-oriented methodologies that guide designers but also users through the process.

Task complexity. We have assumed that the task complexity would increase with the number of items involved. However, during the study, we discovered that the fit of items to the ranking goal can be confounding with respect to task complexity. For 100 items only, participants discovered only very few items that matched the users' preferences for the ranking task. The result was unexpected: users took longer to decide for the set of 20 (weak) items to rank on top. A recommendation would be to design the items for small sample sizes in a way that the task complexity does not increase due to unfortunate value distributions in items.

10. Conclusion

We presented RankASco, a visual analytics approach for the human-centered creation of item rankings. RankASco enables users to interactively express and formalize preferences on attributes, leading to a weighted ranking of items based on multiple scores; one per attribute. RankASco is the result of a two-year research project with multiple design, validation, and reflection iterations. It builds upon a conceptual [8] and a technical [9] workshop paper contribution. We compare RankASco to a general purpose tool in a user study with 24 participants, where users were tasked to create item rankings for an apartment-hunting scenario. The study involved six conditions consisting of the two different ranking approaches and three different dataset sizes. During the study, we also observed 12 variables of user behavior and studied these behaviors with respect to the two approaches (RankASco and Excel). From our observations, we derived three personas as well as guidelines on the applicability of approaches for these personas. Future work includes the expansion of the empirical work to a larger participant group and to the study of more conditions, such as additional ranking approaches.

CRediT authorship contribution statement

Clara-Maria Barth: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing, Visualization, Supervision, Project administration. **Jenny Schmid:** Conceptualization, Methodology, Software, Investigation, Data curation, Writing, Visualization. **Ibrahim Al-Hazwani:** Conceptualization, Writing. **Madhav Sachdeva:** Conceptualization, Software, Data curation, Writing, Visualization. **Lena Cibulski:** Conceptualization, Writing. **Jürgen Bernard:** Conceptualization, Methodology, Validation, Resources, Writing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my data in the paper text.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cag.2023.05.004.

References

[1] Dimara E, Bezerianos A, Dragicevic P. Conceptual and methodological issues in evaluating multidimensional visualizations for decision support. IEEE Trans Vis Comput Graph (TVCG) 2018;24(1):749–59. http://dx.doi.org/10.1109/TVCG.2017.2745138.

[2] Bostandjiev S, O'Donovan J, Höllerer T. TasteWeights: A visual interactive hybrid recommender system. In: ACM conference on recommender systems. ACM; 2012, p. 35–42. http://dx.doi.org/10.1145/2365952.2365964.

[3] Wall E, Das S, Chawla R, Kalidindi B, Brown ET, Endert A. Podium: Ranking data using mixed-initiative visual analytics. IEEE Trans Vis Comput Graph (TVCG) 2018;24(1):288–97. http://dx.doi.org/10.1109/TVCG.2017.2745078.

[4] Kuhlman C, VanValkenburg M, Doherty D, Nurbekova M, Deva G, Phyo Z, Rundensteiner E, Harrison L. Preference-driven interactive ranking system for personalized decision support. In: ACM international conference on information and knowledge management. 2018, p. 1931–4. http://dx.doi.org/10.1145/3269206.3269227.

[5] Pereira MM, Paulovich FV. RankViz: A visualization framework to assist interpretation of Learning to Rank algorithms. Comput Graph Forum (CGF) 2020;93:25–38. http://dx.doi.org/10.1016/j.cag.2020.09.017.

[6] Gratzl S, Lex A, Gehlenborg N, Pfister H, Streit M. LineUp: Visual analysis of multi-attribute rankings. IEEE Trans Vis Comput Graph (TVCG) 2013;19(12):2277–86. http://dx.doi.org/10.1109/TVCG.2013.173.

[7] di Sciascio C, Sabol V, Veas E. uRank: Exploring document recommendations through an interactive user-driven approach. 2015, http://dx.doi.org/10.13140/RG.2.1.5105.0321.

[8] Schmid J, Bernard J. A taxonomy of attribute scoring functions. In: Eurovis workshop on visual analytics (EuroVA). Eurographics; 2021, p. 31–5. http://dx.doi.org/10.2312/eurova.20211095.

[9] Schmid J, Cibulski L, Hazwani IA, Bernard J. RankASco: A Visual Analytics Approach to Leverage Attribute-Based User Preferences for Item Rankings. In: Eurovis workshop on visual analytics (EuroVA). The Eurographics Association; 2022, http://dx.doi.org/10.2312/eurova.20221072.

[10] Nichols D. Implicit rating and filtering. ERCIM; 1998.

[11] Jawaheer G, Szomszor M, Kostkova P. Comparison of implicit and explicit feedback from an online music recommendation service. In: Workshop on information heterogeneity and fusion in recommender systems. 2010, p. 47–51.

[12] Anand SS, Mobasher B. Intelligent techniques for web personalization. Springer; 2005, p. 1–36.
[13] Ricci F, Rokach L, Shapira B. Introduction to recommender systems handbook. In: Recommender systems handbook. Springer; 2011, p. 1–35.
[14] Lu J, Wu D, Mao M, Wang W, Zhang G. Recommender system application developments: a survey. Decis Support Syst 2015;74:12–32.
[15] Liu T-Y, et al. Learning to rank for information retrieval. Found Trends Inf Retr 2009;3(3):225–331.
[16] Bidoki AMZ, Yazdani N. DistanceRank: An intelligent ranking algorithm for web pages. Inf Process Manage 2008;44(2):877–92.
[17] Zehlike M, Bonchi F, Castillo C, Hajian S, Megahed M, Baeza-Yates R. Fa* ir: A fair top-k ranking algorithm. In: ACM on conference on information and knowledge management. 2017, p. 1569–78.
[18] Shneiderman B. Human-centered artificial intelligence: Reliable, safe & trustworthy. Int J Hum Comput Interact 2020;36(6):495–504. http://dx.doi.org/10.1080/10447318.2020.1741118.
[19] Wall E, Das S, Chawla R, Kalidindi B, Brown ET, Endert A. Podium: Ranking data using mixed-initiative visual analytics. IEEE Trans Vis Comput Graph (TVCG) 2018;24(1):288–97. http://dx.doi.org/10.1109/TVCG.2017.2745078.
[20] Edwards W, Barron FH. SMARTS and SMARTER: Improved simple methods for multiattribute utility measurement. Organ Behav Hum Decis Process 1994;60(3):306–25.
[21] Tervonen T, Figueira JR. A survey on stochastic multicriteria acceptability analysis methods. J Multi-Crit Decis Anal 2008;15(1–2):1–14.
[22] Cibulski L, Mitterhofer H, May T, Kohlhammer J. Paved: Pareto front visualization for engineering design. In: Computer graphics forum, vol. 39. Wiley Online Library; 2020, p. 405–16.
[23] Carenini G, Loyd J. ValueCharts: Analyzing linear models expressing preferences and evaluations. In: Working conference on advanced visual interfaces. ACM; 2004, p. 150–7. http://dx.doi.org/10.1145/989863.989885.
[24] Yuan X, Nguyen MX, Chen B, Porter DH. HDR VolVis: High dynamic range volume visualization. IEEE Trans Vis Comput Graphics 2006;12(4):433–45.
[25] Loepp B, Herrmanny K, Ziegler J. Blended recommending: Integrating interactive information filtering and algorithmic recommender techniques. In: ACM conference on human factors in computing systems. 2015, p. 975–84. http://dx.doi.org/10.1145/2702123.2702496.
[26] Pajer S, Streit M, Torsney-Weir T, Spechtenhauser F, Möller T, Piringer H. WeightLifter: Visual weight space exploration for multi-criteria decision making. IEEE Trans Vis Comput Graph (TVCG) 2017;23(1):611–20. http://dx.doi.org/10.1109/TVCG.2016.2598589.
[27] Isenberg T, Isenberg P, Chen J, Sedlmair M, Möller T. A systematic review on the practice of evaluating visualization. IEEE Trans Vis Comput Graph (TVCG) 2013;19(12):2818–27.
[28] Carpendale S. Evaluating information visualizations. In: Information visualization - human-centered issues and perspectives, vol. 4950. Springer; 2008, p. 19–45. http://dx.doi.org/10.1007/978-3-540-70956-5_2.
[29] Pruitt J, Adlin T. The persona lifecycle: Keeping people in mind throughout product design. Elsevier; 2010.
[30] Guo FY, Shamdasani S, Randall B. Creating effective personas for product design: insights from a case study. In: Internationalization, design and global development. Springer; 2011, p. 37–46.
[31] Revella A. Buyer personas: How to gain insight into your customer's expectations, align your marketing strategies, and win more business. John Wiley & Sons; 2015.
[32] Cooper A, Reimann R, Cronin D, Noessel C. About face: the essentials of interaction design. John Wiley & Sons; 2014.
[33] Johansson Fernstad S, Jern M, Johansson J. Interactive quantification of categorical variables in mixed data sets. 2008, p. 3–10. http://dx.doi.org/10.1109/IV.2008.33.
[34] Wang L, Sun G, Wang Y, Ma J, Zhao X, Liang R. AFExplorer: Visual analysis and interactive selection of audio features. Vis Inform 2022;6(1):47–55. http://dx.doi.org/10.1016/j.visinf.2022.02.003.
[35] Kuhlman C, VanValkenburg M, Doherty D, Nurbekova M, Deva G, Phyo Z, Rundensteiner E, Harrison L. Preference-driven interactive ranking system for personalized decision support. 2018, http://dx.doi.org/10.1145/3269206.3269227.
[36] Cheng A, Yin Y, Yan Z, Liu Y, Zhou Z. Visual analytics of multiple network ranking based on structural similarity. In: 2022 IEEE 15th pacific visualization symposium (pacificvis). 2022, p. 196–200. http://dx.doi.org/10.1109/PacificVis53943.2022.00032.
[37] Carenini G, Loyd J. Valuecharts: analyzing linear models expressing preferences and evaluations. In: Working conference on advanced visual interfaces. 2004, p. 150–7.
[38] McHugh ML. The chi-square test of independence. Biochem Med 2013;23(2):143–9.
[39] Patten ML. Understanding research methods: An overview of the essentials. Routledge; 2017.
[40] Wong BD. Bézierkurven: gezeichnet und gerechnet: Ein elementarer zugang und anwendungen. Orell Füssli; 2003.
[41] Wilson ML. Search user interface design. Synthesis lectures on information concepts, retrieval, and services, vol. 3, Morgan & Claypool Publishers; 2011, p. 1–143, (3).
[42] Davis L. Designing a search user interface for a digital library. J Am Soc Inf Sci Technol 2006;57(6):788–91.
[43] Begel A. Codifier: a programmer-centric search user interface. In: Workshop on human-computer interaction and information retrieval. 2007, p. 23–4.
[44] De Pauw J, Ruymbeek K, Goethals B. Modelling users with item metadata for explainable and interactive recommendation. 2022, arXiv preprint arXiv:2207.00350.
[45] Petridis S, Daskalova N, Mennicken S, Way SF, Lamere P, Thom J. TastePaths: Enabling deeper exploration and understanding of personal preferences in recommender systems. In: International conference on intelligent user interfaces. 2022, p. 120–33.
[46] Bernard J, Steiger M, Mittelstädt S, Thum S, Keim D, Kohlhammer J. A survey and task-based quality assessment of static 2D colormaps, vol. 9397. SPIE Press; 2015, 93970M. http://dx.doi.org/10.1117/12.2079841.
[47] Bartelheimer C. Apartment rental offers in Germany. 2020, URL: https://www.kaggle.com/datasets/corrieaar/apartment-rental-offers-in-germany.
[48] Ahlberg C, Shneiderman B. Visual information seeking: tight coupling of dynamic query filters with starfield displays. In: Conference on human factors in computing systems, CHI. ACM; 1994, p. 313–7. http://dx.doi.org/10.1145/191666.191775.
[49] Dimara E, Bezerianos A, Dragicevic P. Narratives in crowdsourced evaluation of visualizations: A double-edged sword? In: CHI conference on human factors in computing systems. 2017, p. 5475–84.
[50] Owen DB. The power of Student's t-test. J Amer Statist Assoc 1965;60(309):320–33.
[51] Hogg RV, Tanis EA, Zimmerman DL. Probability and statistical inference. NJ, USA: Pearson/Prentice Hall Upper Saddle River; 2010.
[52] McKnight PE, Najab J. Mann-whitney u test. In: The corsini encyclopedia of psychology. Wiley Online Library; 2010, p. 1.
[53] Krishnamoorthy K. Handbook of statistical distributions with applications. Chapman and Hall/CRC; 2006.
[54] Kim TK. Understanding one-way ANOVA using conceptual figures. Korean J Anesthesiol 2017;70(1):22–6.
[55] Charmaz K. Constructing grounded theory: A practical guide through qualitative analysis. sage; 2006.
[56] Glaser BG, Strauss AL. The discovery of grounded theory: Strategies for qualitative research. Routledge; 2017.
[57] Brehmer M, Sedlmair M, Ingram S, Munzner T. Visualizing dimensionally-reduced data: Interviews with analysts and a characterization of task sequences. 2014, p. 1–8.