Interactive Explainable Ranking

Chao Zhang
Cornell University
Ithaca, New York, USA
cz468@cornell.edu

Abe Davis
Cornell University
New York, New York, USA
abedavis@cornell.edu

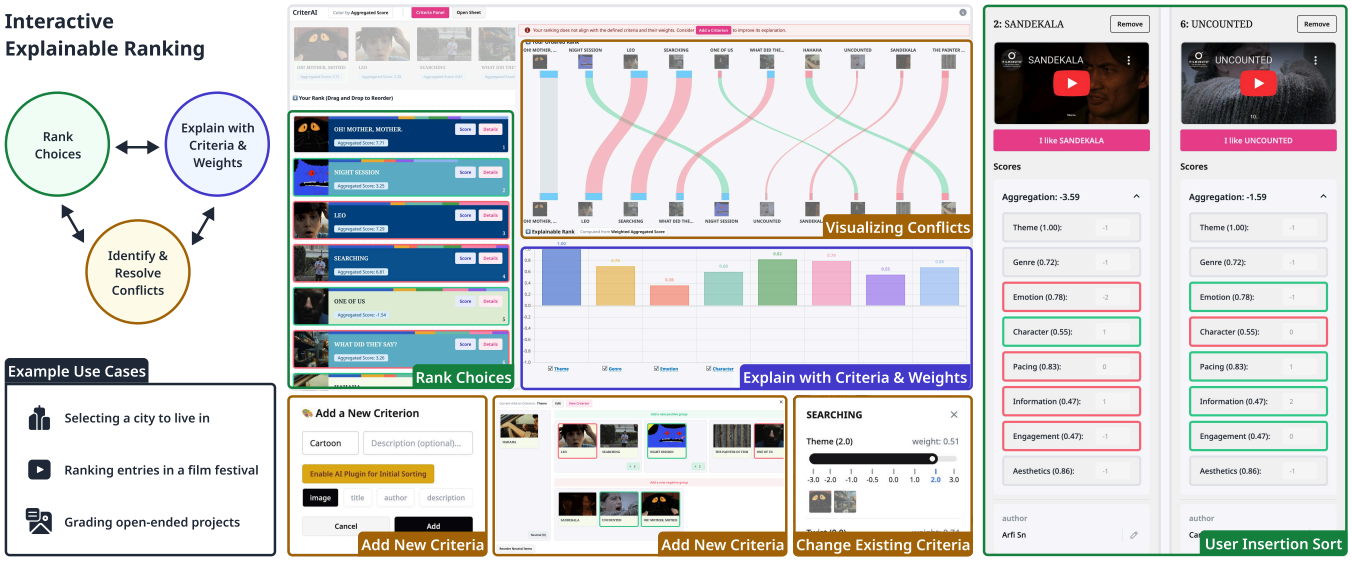

Figure 1: In this paper, we formalize explainable ranking as a new problem for DMTs: given a set of options, the user needs to find a preferred ranking of those options that is consistent with some weighted combination of simpler or less ambiguous criteria (an explanation). To assist users in explainable ranking, our tool makes the three loops interactive: (1) rank choices, (2) explain with criteria and weights, and (3) identify & resolve conflicts. It visualizes conflicts between the user proposed ranking and the ranking explained by current criteria and weights; allows users to freely edit criteria and weights or add new criteria; and offers User Insertion Sort to safely using uncertain priors (e.g., from AI or optimization) while ensuring that every ranking decision is checked by a human user. We evaluate our system on different ranking tasks reflecting real-world use cases.

Abstract

We propose an interactive decision-making tool for discovering and exploring *explainable rankings* for a given set of choices (e.g., job offers, vacation destinations, award candidates). We define an explainable ranking as an ordering of choices based on some consistent weighting of measured criteria. Our tool is designed to help users explore different orderings, criteria, and criterion weights in search of an explainable ranking that reflects their own personal preferences. To achieve this, we combine visualization, optimization, and (optionally) the integration of AI to help users identify and correct or explain inconsistencies in their evaluation of different choices. Through user experiments, we demonstrate that our tool leads to more consistent explainable rankings with greater user confidence.

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/26/04
https://doi.org/10.1145/3772318.3790810

CCS Concepts

• **Human-centered computing** → *Interactive systems and tools.*

Keywords

Ranking, Decision-Making

ACM Reference Format:
Chao Zhang and Abe Davis. 2026. Interactive Explainable Ranking. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3772318.3790810

1 Introduction

Life is full of difficult decisions—ambiguous and multifaceted choices that force us to weigh disparate criteria and balance competing interests. For example, selecting between different academic programs or job offers, evaluating candidates for a position or award, or evaluating potential long-term investments like the purchase of a car or home. The more consequential the decision, the harder it often becomes to make, as every possible ramification presents a new dimension to consider, and cascading impacts may be nuanced, subjective, or uncertain. Navigating these decisions is difficult, which

leaves us susceptible to inconsistencies and cognitive bias, even when a problem is approached with the best intentions [8, 16, 35]. Here, decision-making tools (DMTs) can help by leveraging well-defined criteria to make complexity easier to navigate, and flawed reasoning easier to avoid. But for many decisions, identifying the right criteria is much of the challenge. In this work, we explore a new type of DMT designed to address a general class of decision-making problem we call *explainable ranking*, where the user is given a set of options to evaluate, and their task is to find a preferred ordering of those options (a *ranking*) that is consistent with some weighted combination of simpler or less ambiguous criteria (an *explanation*). What distinguishes our framing of explainable ranking from problems addressed in previous work is the simultaneous exploration of both rankings and explanations, without the constraint that the two should always remain consistent. Previous DMTs have focused on exploring rankings or explanations individually to avoid disagreement between the two. We show that permitting such disagreement allows us to create a more general DMT that can also help users identify and address potential inconsistencies or bias they may not otherwise be aware of. Our work combines design and technical innovation to develop our undertanding of explainable ranking problems and present a novel and practical system to address them in real-world applications.

Overview

Much of our original motivation for this work came from a specific real-world problem: the need to develop fair, consistent, and scalable grading procedures for open-ended projects in a university course. Section 3 describes this problem, how it contributed to our design process, and why many of the challenges it surfaced could not be solved with existing tools. We then propose a set of design goals (Section 4) for explainable ranking tools, including the development of ethical guardrails for the use of automation (e.g., AI and optimization) in evaluations. In Section 5 we formalize explainable ranking to help quantify challenges and show how our framing generalizes beyond open-ended grading. In Section 6, we introduce two key technical innovations that play significant roles in our approach to explainable ranking, and also carry potential significance for other applications:

- *Explanation-Rank Resolution* (Section 6.1): an approach to interactive ranking that offers users unconstrained control over rankings and explanations, and uses a DMT to identify, visualize, and resolve inconsistencies between the two.
- *User Insertion Sort* (Section 6.2): an interactive strategy for safely integrating uncertain priors (e.g., based on AI or optimization) into the ranking process, while ensuring that every ranking decision is checked by a human user.

Section 7 then describes the design of our interactive tool and how it balances our design goals. Finally, in Sections 8-9, we evaluate our system on four different ranking tasks reflecting real-world use cases, including one based on an open-ended grading application that our tool has already been used for to grade hundreds of real student projects over multiple years.

2 Related Work

Decision making has been studied in a wide range of fields, from psychology and economics to HCI, operations management, and game theory. Our formulation of explainable ranking in Section 5 also establishes useful mathematical connections to work in conjoint analysis [6], recommendation systems [18, 43], and Bayesian optimization [42]. In the interest of brevity, we note these wider connections here and discuss mathematical connections as they arise in our method, but focus our the rest of this section on especially relevant ideas from psychology and related work on DMTs in HCI.

2.1 Decision-Making & Cognitive Bias

Large bodies of research have examined inconsistencies and bias in human decision-making [7, 8, 19, 31, 34]. We are particularly interested in what Tversy and Kahnerman first described as cognitive biases and heuristics [34–36]. At a very high level, cognitive biases can be thought of as common mistakes or inconsistencies in human decision-making, and heuristics as theories that predict or explain those mistakes. Two heuristics are especially relevant to our application: availability, and bounded rationality [7, 19, 31]. Essentially, limited memory, knowledge, and reasoning lead humans to take shortcuts when making decisions. One such shortcut is called the anchoring and adjustment effect [35], which describes a bias toward evaluating new options relative to those that have been seen before or recently. For example, a user may be more likely to rate a given option highly immediately after considering several options that are clearly inferior by comparison. For long tasks that call for many comparisons to be made (like ranking) this can result in drifting standards that cause inconsistencies over time. Much of our system design aims to mitigate availability and bounded rationality heuristics with the help of visualization, optimization, and AI.

2.2 Multi-Criteria Ranking Tools

Multi-criteria DMTs have been explored in several previous works from HCI and related fields [9, 22, 25, 33, 40, 41]. The work most closely related to ours comes from systems that focus on ranking tasks [3, 13, 29, 37]. As noted in Section 4.3, these systems can be interpreted as explainable ranking tools where criteria are assumed to be fixed and known a priori, and rankings are directly determined by a weighted combination of those criteria. LineUp [13] allows users to interactively combine attributes and adjust weights to explore how different configurations affect rankings. RankASco [3, 29] visualizes categorical and numerical attributes along with underlying correlations, and lets users explore non-linear weighting schemes for different criteria. Podium [37] lets users specify a set of soft ranking constraints (i.e., an ordering for some subset of choices) and then optimizes attribute weights to best fit those constraints, which, to our knowledge, is the closest that previous work has come to allowing independent exploration of rankings and criteria. These systems offer a powerful way to explore rankings when each choice can be summarized completely by its provided attributes. However, this setting often fails to adapt to more nuanced or subjective comparisons, and as the number of possible choices grows, so does the portion of possible rankings that cannot be explained with fixed criteria. We formalize these limitations in Section 5.

2.3 AI-Assisted Decision-Making

HCI researchers have increasingly worked on incorporating AI into decision-making systems, for example, by providing humans with recommendations [30]. However, concerns about algorithmic bias and its potential to lead to unfair or harmful outcomes are significant [10, 20], leading to significant interest in research on algorithmic fairness [1], transparency [24, 27, 38, 39], accountability, and appropriate reliance [4, 5, 14, 23]. Recent work by Echterhoff et al. [11] takes a somewhat different approach. Like us, they focus on using AI to find inconsistencies in the user's decisions, specifically by capturing and balancing anchoring bias in sequential decision tasks. We believe this strategy for leveraging AI has a lot of yet unexplored potential.

3 Formative Exploration: Open-Ended Grading

Early motivation for this work came from a specific use case related to scaling fair grading procedures for open-ended projects in a university course on computer graphics, which we will refer to in this paper as *CourseX*. The course is designed to center around open-ended projects that encourage students to use course concepts in creative ways. Three of these projects are assigned over the course of the semester. As enrollment in the course has grown, so has the number of projects to grade. In the six offerings that have used this project-based structure, enrollment has ranged from 84 to 300 students, with most semesters averaging around 145 students[1]. A majority of the projects were done in pairs, so most semesters have called for grading 3 projects with around 70 submissions each.

3.1 Open-Ended Grading & Explainable Ranking

3.1.1 Open-Ended Projects vs. Subjective Criteria. It is useful to distinguish between what we call open-ended projects and assignments that simply include subjective grading criteria. Our work is relevant to both, but open-endedness is primarily responsible for challenges related to scaling. Subjective grading criteria are quite common, even in large classes. For example, in liberal arts coursework, writing assignments often necessitate some subjectivity in grading. However, subjective grading criteria can still be carefully specified, for example, in a rubric and/or through examples provided for each achievable grade. When we refer to "open-ended" projects, we mean ones where students are incentivized to go beyond provided specifications in creative and often surprising ways. Projects can be open-ended without having subjective criteria. For example, imagine asking students to build a mechanism to launch tennis balls as high as possible into the air. Even if the criterion for grading is objective (e.g., the altitude reached by a ball), the lack of a known upper bound means that, if we grade on a curve, there is a strong incentive for students to find creative new solutions to the problem (literally, to shoot for the moon). This example also illustrates the need for ranking: if the distribution of criteria is unbounded and unknown when the assignment is given, the mapping from projects to grades cannot be determined until all projects have been evaluated—at which point it becomes a ranking problem.

[1]Enrollments fluctuate over the semester, so these numbers are approximate. Variation across offerings stabilized after a cap of 150 was placed on enrollment.

When multiple criteria are used, even if each is individually objective, graders are tasked with determining their respective weights. When the criteria are known and fixed, this becomes the problem addressed by Multi-criteria DMTs. Explainable ranking generalizes this further by helping users incorporate new and subjective criteria during the ranking process.

3.1.2 Rubrics & Grading Criteria. Open-ended projects and grading do not preclude the use of rubrics and well-specified grading criteria. In *CourseX*, for example, rubric with criteria for grading each project are provided when the project is assigned, with specific guidelines on what is required to achieve grades up to a given threshold (roughly in the B range). Grades up to this threshold are uncurved, and based on the provided rubric. However, grades above the threshold are differentiated on a curve, which is what leads to ranking. While somewhat uncommon for grading, this scheme is similar to many decision problems encountered in other settings. For example, when making hiring decisions, employers often use a set of requirements to filter candidates before ranking those that remain. Filtering is much faster than ranking, because candidates only need to be evaluated against a single list of criteria, while ranking requires evaluation against all other candidates. Rubrics can serve a related purpose in grading for large courses: the effort of evaluating a single assignment against a rubric does not depend on the number of students in a course, but effort required to rank an assignment does.

3.2 Inconsistency in Ranking

3.2.1 Local vs. Global Assessment. When criteria for ranking are subjective, nuanced, or varied, this gives rise to two distinct types of assessment. The first type, which we call *local* assessment, involves evaluating an item in isolation to understand its merits and how they relate to existing criteria. Grading based on well-defined rubrics amounts to local-only assessment. However, ranking allows for a second type of *global* assessment, where items are compared to weigh their relative merits. Global assessment cannot precede local assessment, as one cannot compare the merits of two items without first understanding what those merits are. With this in mind, the first step in our approach to grading involves evaluating submissions according to the provided rubric. This determines the uncurved portion of grades, serves as a way to identify the strengths and weaknesses of each submission.

3.2.2 Estimating Consistency in Local Assessments. A common strategy for scaling grading is to have several graders work in parallel. We also used this strategy to scale the local assessment of submissions in our grading. We begin by grading a sampling of submissions as a group in the same room for calibration. Then, the remaining submissions are divided among graders such that each submission is graded by two graders. Initial scores are then calculated for each submission by averaging its two corresponding scores. In our first iteration of the course, grading ended here, and we later discovered several inconsistencies between grades assigned to different projects. That discovery was the original motivation for this work, eventually leading to the system we use now, which follows an initial round of parallel local assessments with a round of global assessments performed with our interactive tool as a group, and

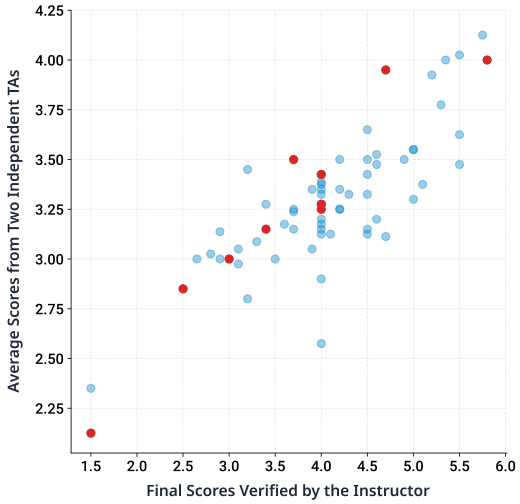

Figure 2: The x-axis shows the final scores that are verified by the instructor. The y-axis shows first pass scores that are the average of values assigned independently by two TAs. Red dots indicate the projects we used in our second case study.

a final round of global assessments by the instructor the next day to check for consistency. Finally, the instructor applies a curve to the verified rankings to determine project grades. This new process offers us a lens into the consistency of local assessments by letting us compare initial scores against the final scores that are verified by the instructor. Figure 2 shows anonymized data from a project we refer to as C2, where final scores are plotted against the initial local estimates by course staff. The Spearman's rank correlation between the initial and final scores was 0.74 ($p < 0.001^{***}$). From Figure 2, we see a clear but noisy correlation between first pass and final rankings. However, if we measure this noise in terms of "bad" grading decisions—i.e., those where the ordering of two grades was deemed inconsistent or incorrect upon further scrutiny—the results suggest that simply averaging multiple graders is quite problematic on a sufficiently open-ended project. In practical terms, had we used initial scores to assign grades (as many courses do!), most of the students in the class would be able to find several submissions that received higher scores than their own for what most of the course staff would agree to be weaker work. In our second case study (Section 9.2), we examine this grading application in more detail with four of the original graders from the course to better examine how our tool helps make their evaluations more consistent.

3.3 Grading vs. General Explainable Ranking

Our grading application is representative of many common decision problems that can be expressed as explainable ranking. What makes these problems difficult is balancing explainability with the need to compare many diverse items based on diverse and sometimes unanticipated criteria. This challenge is, in essence, one of generalization. It should be no surprise, then, that an effective solution would generalize to other problems. For example, to evaluating award nominees, comparing job applicants or offers, or navigating important purchases. To facilitate such generalization, we derive design goals and a mathematical formulation of the problem based on explainable ranking more generally, which we formalize in Section 5. We later evaluate our tool on three scenarios in addition to grading in user and case studies, and include more examples in our supplemental material.

4 Design Goals

Based on our experiences with open-ended grading and through iterative testing and design, we derived four high-level goals for explainable ranking tools: explainability, adaptability, scalability, and ethical guardrails. Our major design challenge is to balance these goals when they are often in conflict.

4.1 Explainability

Humans are notoriously bad at making complex multi-faceted choices. The core idea of explainability is to factor those choices into a combination of much simpler and less ambiguous decisions that can be easily explained. This serves two purposes: for the user, it supports metacognitive awareness by externalizing their thought process, often making it easier to recognize inconsistencies or bias. And for others, it can offer transparency into the user's reasoning as a way to provide greater insight and accountability. A common example of this can be found in the use of rubrics for scoring participants in a competition. Such rubrics help judges score each competitor fairly, and help competitors understand why they have received a given score. Similarly, our system should help users evaluate choices consistently, while also generating an interpretable explanation for their final ranking. In our testing with users, we measure this by evaluating user confidence in rankings, and the consistency of those rankings. Here, consistency can be measured by testing whether subsequent pairwise comparisons agree with earlier ranking decisions. Note that this kind of consistency implies a weak form of explainability even in isolation, as it offers evidence that a ranking has a valid explanation, even if that explanation has not been made explicit.

4.2 Adaptability

A common limitation of rubrics is that they fail to adapt to less structured tasks. For example, when used for grading, it may be simple to assign values to different answers for a multiple-choice problem, but much more complicated for essays or other more open-ended tasks. This is where adaptability becomes essential. A good example of adaptability can be found in one of the oldest known DMTs, the pros and cons list, which we can think of as an explainable ranking tool for binary decisions (e.g., *A* vs. *B*) and binary criteria values ("pro" or "con"). By limiting choices and criteria to binary value, the pros and cons list minimizes the complexity of adding new criteria, which lets the user focus on enumerating and weighing different factors that may impact their decision. Similarly, we want our system to make it easy for users to add and customize new criteria—but in our case, without limiting choices and criteria to binary values.

4.3 Scalability

The pros and cons list is very adaptable, but limited binary decisions. Decisional balance sheets [17], which extend the pros and cons list to non-binary choices, are often used in psychological interventions

[15], but rarely to decide between more than a very small number of options (e.g., 2-3) for reasons we discuss in Section 5.2. More recent work on multi-criteria DMTs for ranking trades adaptability in favor of scalability by limiting criteria to qualities that are fixed and known [3, 13, 29, 37]. This reduces the problem to weighing known criteria, which maps well to computation even when the number of options becomes large. Our work aims to balance this kind of scalability with more adaptable control over criteria. Here, the goal need not be to scale our process to hundreds or thousands of options—though we will discuss how this might be accomplished as future work—rather, even scaling to tens of options constitutes while maintaining adaptability is significant.

4.4 Ethical Guardrails

It is important to understand that explainability is not the same as fairness. It can often improve fairness by helping users avoid certain types of bias or inconsistency—this is one of the main motivations for our work—but that does not preclude self-consistent biases (e.g., use of deliberately biased criteria), nor does it prevent a determined user from using bad-faith explanations to justify biased decisions. This raises two particular concerns that influence how we approach the integration of AI and optimization in our system:

- **Bias in AI:** Our system supports the use of AI for scalability. However, AI is subject to its own biases [10, 20], and in some cases, its use for evaluation may even be inherently unethical.
- **Explanations vs. Excuses:** Our system supports the use of optimization to help users reason about the values implied by a given ranking. As we discuss later in the paper, such optimization could potentially be abused to find plausible explanations for decisions made in bad faith.

These concerns can turn effective tools for explainable ranking into something of a double-edged sword. With this in mind, our fourth design goal is to limit the risk of potential abuse, which we do by applying two principles to our design:

(1) Rather than using AI or optimization to perform evaluations automatically, we focus on using them to accelerate human evaluation or identify inconsistencies in a users' reasoning. A key strategy we use to accomplish this is what we call *user insertion sort*, described in Section 6.2.
(2) We make uses of AI and optimization optional and modular so that they can be restricted or disabled when appropriate.

5 Problem Formulation & Strategies

Here, we establish a formal definition of explainable ranking, which we can use to quantify the strengths and limitations of existing tools, and to guide the design of an alternative approach.

5.1 Formulation

The user is given some set of n choices, $X = \{x_1, ..., x_n\}$, to compare. The goal is to help them find an *explainable ranking* of these choices, which we will define as one that is consistent with some linear combination of known attributes. More formally, an explainable ranking requires three things:

- A *ranking* $\preceq_*$ that gives a partial ordering of the choices being evaluated, with $x_i \preceq_* x_j$ indicating that x_j is valued as greater than or equal to x_i
- A *criteria matrix* $\mathbf{X}$ with columns $\{\mathbf{x}_0, \cdots, \mathbf{x}_n\}$, where each $\mathbf{x}_i$ is a vector of k values describing the corresponding attributes of choice x_i
- A *weight vector* $\mathbf{w}$ that describes the relative importance of each measured attribute.

We say that the ranking $\preceq_*$ can be *explained* by $\mathbf{X}$ and $\mathbf{w}$ if its ordering of X is consistent with values of the product $\mathbf{w}^\intercal\mathbf{X}$. More precisely, if we define $\preceq_\mathbf{w}$ as:

$$x_i \preceq_\mathbf{w} x_j \iff \mathbf{w}^\intercal\mathbf{x}_i \leq \mathbf{w}^\intercal\mathbf{x}_j \tag{1}$$

then $\preceq_*$ is *explained* by the product $\mathbf{w}^\intercal\mathbf{X}$ when:

$$x_i \preceq_* x_j \implies x_i \preceq_\mathbf{w} x_j \tag{2}$$

Intuitively, if we think of $\mathbf{w}^\intercal\mathbf{x}_i$ as values for x_i predicted by $\mathbf{X}$ and $\mathbf{w}$, then Eq. 2 states that $\preceq_*$ should rank items consistent with these predictions. Our task is then to help the user find or build $\{\preceq_*, \mathbf{X}, \mathbf{w}\}$ that best reflect their own evaluation of the items in X.

5.2 Existing DMTs for Constrained Explainable Ranking

Existing tools had addressed special cases of the explainable ranking problem under different constraints, which are useful to interpret in terms of our design goals.

5.2.1 Pros and Cons List. The classic pros and cons list favors adaptability over scalability by limiting the number of choices to just two options: one supporting a decision ("Pro") and the other rejecting it ("Con"). Each item in the list represents a criterion row of $\mathbf{X}$, where "pro" items can be encoded as rows with 1 in the Pro column and 0 in the Con column, and "con" items encoded as rows with the opposite values. Interestingly, so long as our list has at least one pro and one con, all possible rankings (Pro, Con, or tie) can be explained. As a consequence, the weight vector $\mathbf{w}$ can be wholly implied, as any decision can be defended after the fact with an appropriate choice of weights. This makes adding a criterion to $\mathbf{X}$ extremely easy, as the user only needs to consider two things: whether that criterion is positive or negative, and whether it changes their final decision.

5.2.2 Previous Multi-Criteria DMTs. Most existing multi-criteria DMTs represent a near-opposite approach to the pros and cons list, focusing on scalability at the expense of adaptability. This is typically achieved by limiting the user to interactions that change the weight vector $\mathbf{w}$. The criteria matrix $\mathbf{X}$ is assumed to be given and immutable, and the ranking $\preceq_*$ is set to a direct function of $\mathbf{w}^\intercal\mathbf{X}$ (i.e., $\preceq_* = \preceq_\mathbf{w}$), effectively limiting the user to rankings that are explainable by construction. In this case, every edit to $\mathbf{w}$ triggers the calculation of $\mathbf{w}^\intercal\mathbf{X}$ and a sorting operation, which is easy to compute at interactive rates even for very large criteria matrices. This lets the user focus on exploring the space of possible criteria weights. Notably, Podium [37] deviates from this slightly by letting the user estimate weights based on a set of ordering constraints. This is a step toward some of the interaction that our tool supports, but as $\mathbf{X}$ is immutable, user-specified ordering constraints can often be impossible to satisfy.

6 Key Innovations

Our approach to explainable ranking introduces two innovations that feature significantly in the design of our tool:

6.1 Explanation-Rank Resolution

Existing tools manage the complexity of explainable ranking by prohibiting direct control over one or more of the parameters in $\{\preceq_*, \mathbf{X}, \mathbf{w}\}$. This prevents conflicts from arising between rankings and explanations, but also limits adaptability or scalability. Our tool takes a different approach by letting users edit any part of $\{\preceq_*, \mathbf{X}, \mathbf{w}\}$ independently, making it highly adaptable by design. Notably, doing so permits edits that can put the current ranking $\preceq_*$ in conflict with the current explanation $\mathbf{w}^\intercal\mathbf{X}$. This is one of our biggest departures from previous approaches: rather than forcing rankings to be explainable by construction, our system focuses on helping users identify and resolve contradictions as they arise. To describe this process, which we call *explanation-rank resolution* (ERR), we must distinguish between the current ranking $\preceq_*$ and the one suggested by the current explanation, $\preceq_{\mathbf{w}}$. Conflicts between $\preceq_*$ and $\mathbf{w}^\intercal\mathbf{X}$ often signify inconsistent reasoning or missing criteria. This is both an opportunity and a challenge for ERR, because if we can navigate the added complexity, we can help them identify, visualize, and resolve potential flaws in their own reasoning.

6.2 User Insertion Sort

Many of the individual tasks involved in ERR can be expressed as a sorting operation on choices according to different criteria. For example, the ranking task itself is a kind of sorting, as is the task of checking ranks or criteria values for consistency, or updating $\preceq_{\mathbf{w}}$ based on changes to $\mathbf{w}^\intercal\mathbf{X}$. For completely objective criteria, this process is simple to automate, but for subjective criteria, reliable sorting depends on asking the user to make a series of pairwise comparisons [32]. One of the most powerful techniques we introduce in our tool is what we call *user insertion sort* (UIS), as it offers a way to safely leverage incomplete or uncertain information to accelerate this kind of subjective sorting. The basic idea follows the observation that the insertion sort algorithm 1) can be conducted with pairwise comparisons, and 2) has time complexity that depends heavily on the insertion order. Notably, the number of comparisons used in insertion sort reduces to the same minimum number required to manually verify that an ordering is correct. UIS works by asking the user a sequence of pairwise comparisons that map to the operations of insertion sort, where the insertion order can be determined by an imperfect guess at the true ordering. For good guesses, this accelerates the sorting process to a linear number of comparisons, while ensuring that every decision is checked by the user. In our implementation, we also randomize the order that items appear on-screen to help minimize ordering bias.

There are two main uses for UIS in our tool. First, we let users trigger a sort to set $\preceq_*$ based on the order $\preceq_{\mathbf{w}}$ predicted by current weights to check and see if the current explanation is consistent with their subjective ranking of choices. Second, UIS offers a potentially safer way to incorporate AI and optimization into evaluations. More specifically, when assigning values to criteria, we can use values estimated by AI to determine an insertion order for UIS. This offers three options for how to integrate AI into the system: we can disallow it completely in high-risk tasks, we can use it to fully automate the evaluation of criteria when doing so carries little risk, or we can use it in UIS to accelerate manual evaluation of criteria.

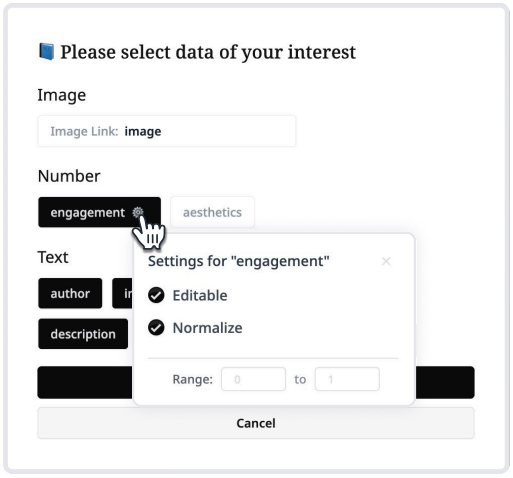

Figure 3: *Data Loading Panel.* Users select which fields to import (image, numeric, text) and configure numeric columns' editability and normalization range.

7 System Design

We begin by describing the data format our tool supports. Then we introduce the different parts of our interface and the typical use patterns they supports. Our strategy for balancing adaptability and scalability will rest on offering a flexible and efficient way to resolve conflicts between $\preceq_*$ and $\preceq_{\mathbf{w}}$ (i.e., ERR). Section 7.3 will discuss how this is done without optimization and AI and when doing so becomes difficult to scale. Section 7.4 then describes how we integrate optimization and AI to address these problems with scalability.

7.1 Data Format and Loading

Our tool supports a wide range of data types, including text, numbers, images, files (e.g., PDFs), and videos. Users prepare data in a Google Sheet where, starting at the third row, each row represents a choice. The first row contains the field names (column headers), and the second row specifies the field types. We support six types: `image` (a link to an image: JPG/PNG/SVG/GIF), `info` (free-form text describing the choice), `name` (the display name for each choice), `file` (a link to a readable file stored in Google Drive, e.g., a PDF), `criterion` (numeric information for a choice, such as pre-scored criteria), and `video` (a link to a YouTube video).

To upload data, users paste the Google Sheet link into our tool. The **Data Loading Panel** (Figure 3) then opens, showing each field as a toggle. Users select which fields to include in the tool. For numeric fields, users can also choose whether the values are editable and whether to normalize them on import (and, if so, specify the normalization range).

Figure 4: ***User Interface.*** (A) Target Rank Panel: Drag items to edit their target ranking. Green borders mark items placed lower than their position in the current explanation; red borders mark items placed higher. (B) Weights Panel: Each criterion's weight is shown as a bar; drag to adjust. Click Estimate Weights (K) to run an SVM that infers a weight vector from the target ranking. (C) Weight Rank Panel: A horizontally scrollable list ordered by the current explanation; not directly editable because it is a function of $\mathbf{w}$ and $\mathbf{X}$. (D) Rank Comparison Panel: A slope chart showing conflicts between the target rank (top) and the explainable rank (bottom). Clicking an item reveals dots at intersections; clicking a dot (G) opens the Choice Comparison Panel (Figure 5). Use the Switcher (H) to switch to the Score Distribution Panel (Figure 7). (F) Scoring Panel: Each criterion is a slider; as users drag, all choices sharing the current score on that criterion are shown for comparison. (E) Criteria Panel Button: Opens the Criteria Panel (Figure 6). (I) Add Criterion Button: Opens the Adding New Criteria Panel (Figure 8). (J) User Insertion Sort Button: Initiates UIS, prompting users to make a series of pairwise comparisons in the choice comparison panel (Figure 5) to rank all choices.

7.2 Interface

Figure 4 shows the main page of our interface. With n choices and k criteria, the $k \times n$ criteria matrix $\mathbf{X}$ is the biggest obstacle to scalability. With this in mind, the main page of our interface focuses on editing $\preceq_*$ and $\mathbf{w}$, and on visualizing inconsistencies between $\preceq_*$ and $\preceq_{\mathbf{w}}$, while operations that deal with more detailed information about $\mathbf{X}$ are abstracted into separate panels that can be toggled as needed. Our main page has four different panels:

- The **Target Rank Panel** (Figure 4A) at the bottom left represents $\preceq_*$ with an ordered list of choices. Users can drag and drop items in the list to edit $\preceq_*$.
- The **Weights Panel** (Figure 4B) at the bottom right represents $\mathbf{w}$, with each criterion weight shown as a different bar. Users can click and drag on these bars to increase or decrease their value.
- The **Weight Rank Panel** (Figure 4C) at the top left shows a small horizontally scrollable list of choices ordered according to the current explanation $\preceq_{\mathbf{w}}$. This ordering cannot be edited directly, as it is a function of $\mathbf{w}$ and $\mathbf{X}$.
- The **Rank Comparison Panel** (Figure 4D) at the top right visualizes conflicts between the current ranking $\preceq_*$ and the current explanation ranking $\preceq_{\mathbf{w}}$ in a vertical slope chart (see Section 7.3).

Actions by the user can also toggle one of three additional panels:

- The **Criteria Panel** (Figure 6), toggled on the main page (Figure 4E), is used to define or edit criteria. This is arguably the single most difficult operation to scale, and the main place where we explore the (optional) integration of AI.
- The **Scoring Panel** (Figure 4F), toggled by selecting a choice in the Target Rank Panel, lets the user view and edit criteria scores for a particular choice, with each criteria represented as a slider.
- The **Choice Comparison Panel** (Figure 5) shows info and criteria scores for specific choices, aligned next to one another to facilitate detailed comparisons and edits.

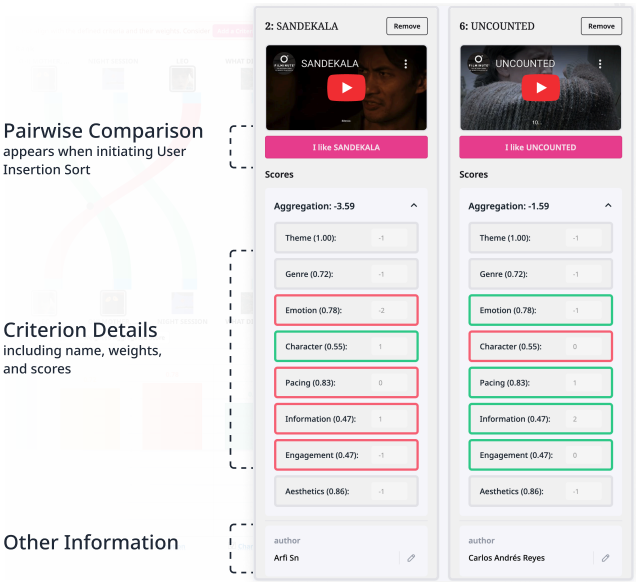

Figure 5: *Choice Comparison Panel.* This panel shows info for two choices, aligned next to one another to facilitate comparisons. It is toggled by clicking the dots at the intersections of conflicts in the Rank Comparison Panel or when users initiate UIS.

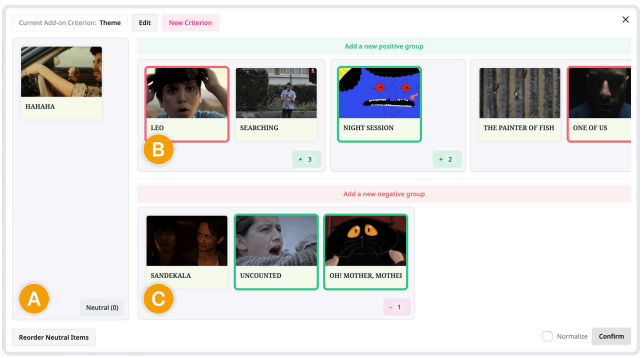

Figure 6: *Criteria Panel.* This panel is divided into three bins representing neutral (A), positive (B), and negative (C) criterion values. Initially, all choices appear in the neutral bin, ordered by predicted values. The positive and negative bins contain groups for different possible scores, and users can add new groups to introduce additional score levels. Users assign attribute values by dragging choices into the appropriate groups.

The most common pattern of use involves editing one of the three components $\{\preceq_*, \mathbf{X}, \mathbf{w}\}$, visualizing the impact of that edit, and then exploring ways to resolve any inconsistencies that are created.

7.3 Viewing & Resolving Contradictions

7.3.1 Visualizing Contradictions. Whenever the current ranking is inconsistent with the current explanation, a warning will show at the top of the page, and contradictions between $\preceq_*$ and $\preceq_{\mathbf{w}}$ will be visible in the Rank Comparison Panel, which displays the ranking given by $\preceq_*$ in its top row, and the one given by $\preceq_{\mathbf{w}}$ in its bottom row. Choices ranked the same by both are connected with a vertical gray line, while those favored by $\preceq_*$ are connected with a sloped green line, and those favored by $\preceq_{\mathbf{w}}$ are connected with a sloped red line. In this visualization, each intersection of two connecting lines indicates a direct contradiction in how $\preceq_*$ and $\preceq_{\mathbf{w}}$ order a given pair of choices. If the user clicks on a given choice in this visualization, small buttons will appear at each intersection involving that choice, and if the user clicks on one of those buttons, the two choices involved in its corresponding contradiction will be loaded into a new Choice Comparison Panel (Figure 5). This makes it easy to compare detailed views of the two choices to determine the nature of the contradiction.

In addition to the slope chart, the Rank Comparison Panel provides two stacked bar charts that visualize the score distributions across choices and criteria for the user's target rank and the current weight rank, respectively (Figure 7). Users can compare these distributions to identify which criterion contributes most to the inconsistencies. These stacked bar charts can be toggled using the Switcher in the corner of the Rank Comparison Panel (Figure 4H). All visualizations update in realtime in response to any edit the user makes, offering immediate and interactive feedback to the user at all times.

7.3.2 Resolving Contradictions Using Existing Criteria. Users can resolve conflicts in several ways. First, they can edit $\preceq_*$ in the Target Rank Panel or $\mathbf{w}$ in the Weights Panel. However, the ranking $\preceq_*$ can easily become difficult or impossible to explain given a fixed set of criteria (e.g., because those criteria fail to capture the value of some choices). In this case, the user may want to create or edit criteria in $\mathbf{X}$. Changes to existing criteria are easiest to make through the Scoring Panel or the Choice Comparison Panel. The sliders in the Scoring Panel (see Figure 4F) are quantized to values held by existing choices, and as the user adjusts a slider, it shows all of the current choices that share the current score for the corresponding criterion. This effectively turns the slider decision into a sequence of pairwise comparisons that tend to be more consistent [32].

7.3.3 Adding New Criteria. Users can also address contradictions or deficiencies in the current explanation by adding new criteria through the Criteria Panel (Figure 6). When a user clicks the create new criterion button on the main screen, they are first prompted to provide basic information, including a name for the new criterion, to the panel shown in Figure 8 (more on this later). They are then presented with the Criteria Panel, which is designed to help specify the scale of possible criteria values used in the Scoring Panel, and to initialize each choices with a value from that scale. The Criteria Panel is split into three bins corresponding to neutral (left), positive (top right), and negative (bottom right) criterion values. Initially, all choices are assigned to the neutral bin, where they are ordered according to value predictions we will discuss later. The positive and negative bins each contain groups that correspond to the different possible scores for the criterion, and the user can add new groups to allow new possible scores. The user can then assigns attribute values by dragging choices to the groups with those corresponding values.

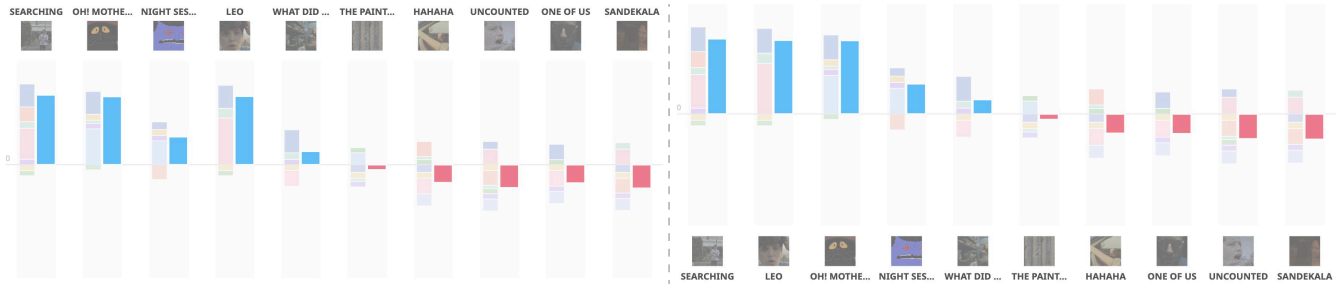

Figure 7: *Score Distribution Panel.* Two stacked-bar charts visualize the score distributions across choices and criteria for the user's target rank (left) and the current weight rank (right). Within each chart, choices are ordered by the respective rank and shown with pairs of columns: the left column shows stacked per-criterion contributions (positive above zero, negative below), and the right column shows the resulting aggregated score.

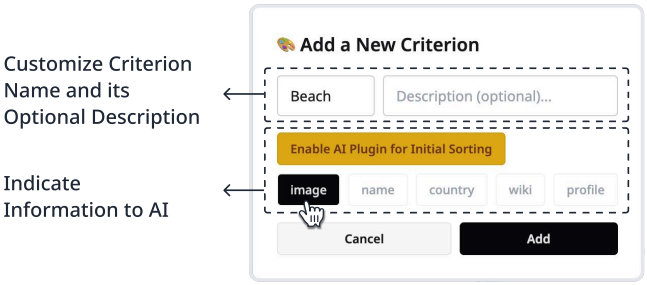

Figure 8: *Adding New Criteria and Optional AI Configurations.* When users click the "Add a Criterion" button on the main page, this panel appears, allowing them to specify the name and description of a new criterion. Users can optionally enable AI to estimate an initial ordering of choices by selecting which aspects (images or text) the AI should consider. Images are used to compute CLIP scores, while textual information is processed by GPT-4o to determine whether the choice satisfies the user-defined criterion.

7.4 Integrating AI & Optimization

7.4.1 Integrating AI models. The primary way that we incorporate AI in our design is through the use of different AI models to estimate values for new criteria. This can be done in two ways depending on ethical concerns. The simplest way is to just take the AI model's criterion values and use them directly. The second way is to use them to derive an insertion order for UIS. When enabled, our current implementation offers estimates based on CLIP [28] or GPT-4o [26]. When the user creates a new criteria they indicate what information they expect to be most predictive of that criteria and provide a query question or term for the model to use (Figure 8). If users indicate that the criteria is related to an image, we sort by CLIP cosine similarity with the query to determine insertion order. If the user indicates that a text-based property from the choice descriptions is relevant, then the system will determine insertion order by sorting GPT-4's responses to the query given the corresponding property text for each choice (Figure 9).

7.4.2 Integrating Optimization with SVMs. For a given ranking $\preceq_*$ and criteria matrix $\mathbf{X}$, we can use optimization to infer the weight vector $\mathbf{w}$ that explains $\preceq_*$ by the largest margin (i.e., farthest from having ties). This can be done by training a support vector machine (SVM) to classify the difference $\mathbf{x}_i - \mathbf{x}_j$ between two criterion vectors according to whether $\mathbf{x}_i \preceq_* \mathbf{x}_j$. Crucially, for certain $\preceq_*$ and $\mathbf{X}$, this margin of this optimization is negative, indicating that no set of weights can explain $\preceq_*$ in terms of $\mathbf{X}$. We can use this SVM strategy to solve for weights that justify a given ranking, or to determine when no such weights exist. This can be a useful tool for introspection, as it lets the user rank items based on more subjective intuition, and then solve for weights that better quantify that intuition. This idea has been developed independently in a number of different fields, including for recommendation systems [18, 43], conjoint analysis [6], and most recently to formulate order-based constraints in the related DMT Podium [37]. However, this application of SVMs becomes even more useful when we combine it with UIS and the ability to add new criteria.

Given a set of weights that fails to explain the current ranking (e.g., from an SVM that fails to classify with non-negative margin). We can interpret each contradiction as a support vector with negative margin. If we assume that this negative margin is the result of a single missing criterion, then this margin provides a powerful way to predict the value of this missing criterion. Intuitively, this is analogous to saying that if the user is adding a new criterion to better capture a target $\preceq_*$, then contradictions between $\preceq_*$ and the current $\preceq_{\mathbf{w}}$ can tell us whether the choices involved should have positive or negative values for this criterion. We use this to estimate our default insertion order for new criteria when no other estimate is given.

7.5 Implementation Notes

The frontend of our tool is built with Next.js, leveraging server-side rendering to manage API calls. The drag-and-drop interaction is implemented using react-sortablejs[2]. The interactive bar chart for weights is created with react-chartjs-2[3]. The visualizations in the Rank Comparison Panel are rendered using SVG elements. The

[2]https://www.npmjs.com/package/react-sortablejs
[3]https://react-chartjs-2.js.org/

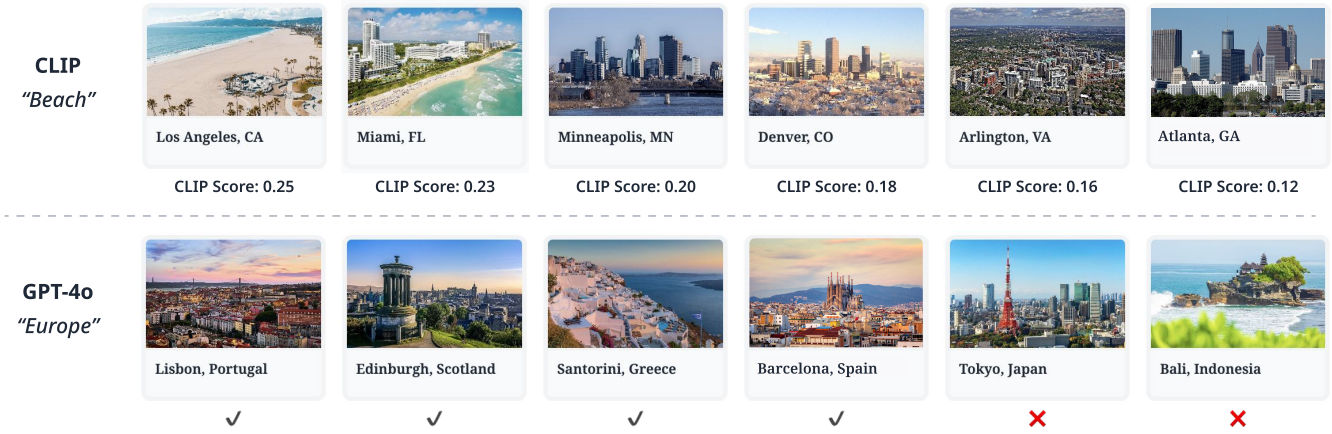

Figure 9: ***Examples of Integrating AI Models to Estimate an Initial Order for a New Criterion.*** The top row sorts choices by CLIP cosine similarity between the image of each choice and the user query "Beach." The second row sorts choices by binary classification results from GPT-4o prompted to determine whether the text of each choice fulfills the user query "Europe."

backend of our tool is built with the FastAPI framework in Python, where we implemented the CLIP pipline and linear SVM classifiier.

8 Study 1: Within-Subjects Study

The within-subjects study aims to examine the usability of our tool and whether it can help users create more consistent, explainable rankings compared to a traditional ranking tool.

8.1 Methodology

This section describes the methodology of this within-subjects study, including the participants and the study procedure.

8.1.1 Participants. We conducted a within-subjects experiment with 8 participants (3 female, 5 male; P01—P08) aged 23—27 (M = 24.5, SD = 1.20), recruited through message boards. Each session lasted approximately 60 minutes, and participants were compensated with $20 USD for their time.

8.1.2 Procedure. After obtaining informed consent and collecting demographic information, participants completed two ranking tasks, each under a different condition, with task order counterbalanced to mitigate order effects.

Tasks: In one task, they ranked 12 cities to live in, with cost-of-living information provided as the only initial criteria, so users defined additional criteria as they progressed. In the other task, they ranked 12 very short (~5s) videos as if judging entries for a rendering challenge. Here, they were given the number of YouTube views as the only starting criteria and developed additional criteria based on personal preferences. Each city or video was presented with a brief textual description and a link (e.g., Wikipedia or video page) for further reference. Participants were also allowed to search online freely to gather additional information as they developed their ranking criteria, to better represent real-world use. They could identify key factors, assign scores, and refine their rankings based on their own reasoning.

Baseline: For the baseline condition, participants used Google Sheets, where the initial information was pre-filled. We ensured that participants knew how to use formulae to compute explanations in Google Sheets (i.e., calculating a weighted sum of criteria). Because all participants were already familiar with this tool, we did not provide a tutorial. We selected Google Sheets as the baseline because the general nature of our task makes most existing DMTs inapplicable (see Section 5.2), and Google Sheets is one of the most widely used tools for organizing and ranking items. Its flexibility allows users to manually perform explainable ranking actions, such as adding columns to represent additional criteria, reorganizing their rankings by moving items, and computing weighted sums of criteria to sort items by the resulting scores. Before using our tool, we provided a guided walkthrough demonstrating its key features, ensuring participants were familiar with its functionality before beginning the task.

Consistency Check: After completing each task, we administered a consistency check. Participants were presented with five randomly selected item pairs. For each pair, they were shown the basic information of the two items without access to their final ranks or scores. Participants were asked to indicate which item they believed was better in their final rank, provide a brief justification for their choice, and rate their confidence on a 7-point Likert scale.

Post Survey: After each condition, participants completed a survey that included questions about their perceived ease of each actions in explainable ranking. After completing both tasks, participants filled out the UMUX-LITE [21] questionnaire to assess the usability of our system. All survey items used a 7-point Likert scale.

8.2 Results

This section presents the outcomes from the user study. We first computed the usability score from UMUX-LITE. Our system achieved an overall score of 85.42 (SD = 15.27), indicating an excellent usability [2]. Next, we present results from the usage logs, consistency checks, and the post-study survey. Given the ordinal nature of

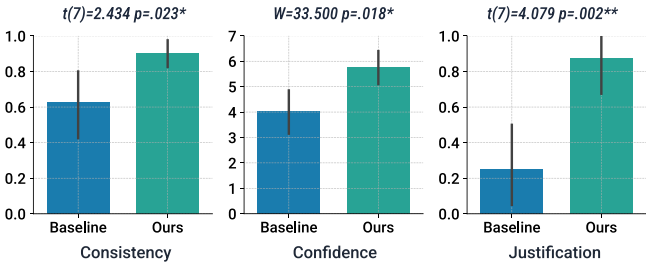

Figure 10: Bar plots illustrating the statistical metrics of participant performance in consistency check under two conditions, where the t-values from the Student's paired t-test, W-values from the Wilcoxon signed-rank paired test, and p-values (*: p<.05, **: p<.01, ***: p<.001) are reported. Error bars represent 95% confidence intervals (CIs).

Likert-scale responses, we employed the Wilcoxon signed-rank test for statistical comparisons. For other quantitative metrics, we used paired t-tests.

8.2.1 Time Spent and Number of Criteria in Ranking Tasks. Participants spent a comparable amount of time (in minutes) across conditions, with no significant difference between our system and the baseline (*ours*: $M = 19.00$ $SD = 3.63$ *vs. baseline*: $M = 14.25$ $SD = 6.21$; $t(7) = 1.984$ $p = 0.088$). However, participants defined significantly more criteria when using our system than with the baseline (*ours*: $M = 5.50$ $SD = 0.93$ *vs. baseline*: $M = 3.13$ $SD = 1.81$; $t(7) = 3.054$ $p = 0.009^{**}$), suggesting that our interface encouraged richer and more deliberate articulation of ranking dimensions.

8.2.2 Consistency Check. To assess internal consistency and participants' ability to explain ranking decisions, we analyzed consistency, confidence, and justification responses from the consistency check task (Figure 10). This aimed to evaluate whether our tool improved the transparency and consistency of participants' reasoning compared to the baseline.

Consistency: For each item pair, we compared the participant's selected item to their original final ranking. If the chosen item had been ranked higher, the response was marked as consistent (1); otherwise, it was marked as inconsistent (0). We then calculated the overall accuracy rate per participant for each condition. The results showed that participants' internal consistency was significantly higher when using our tool compared to the baseline (*ours*: $M = 0.90$ $SD = 0.11$ *vs. baseline*: $M = 0.63$ $SD = 0.29$; $t(7) = 2.434$ $p = 0.023^{*}$).

Confidence: Participants rated their confidence in each pairwise choice. We averaged confidence scores across the five item pairs per condition. Participants reported significantly higher confidence when using our tool compared to the baseline (*ours*: $M = 5.75$ $SD = 1.01$ *vs. baseline*: $M = 4.03$ $SD = 1.33$; $W = 33.500$ $p = 0.018^{*}$).

Justification: We analyzed the justification responses to determine whether each referenced the participant's previously defined criteria or weights. Each justification was coded as aligned (1) if it explicitly referred to those criteria/weights and not aligned (0) otherwise. We intentionally used a strict binary coding to capture only explicit references to participants' defined criteria or weights. This conservative approach avoids subjective judgments of partial alignment and provides a lower-bound estimate of how often participants consciously connected their justifications to their decision framework. The results showed that participants using our tool more frequently referenced their defined criteria in justifying their choices compared to the baseline (*ours*: $M = 0.86$ $SD = 0.28$ *vs. baseline*: $M = 0.25$ $SD = 0.37$; $t(7) = 4.079$ $p = 0.002^{**}$).

8.2.3 Percieved Ease of Explainable Ranking. We examined participants' perceived ease of creating explainable rankings in each condition. As shown in Figure 11, participants reported that understanding the explainability of their rankings was easier with our tool compared to the baseline (*ours*: $M = 6.00$ $SD = 1.41$ *vs. baseline*: $M = 3.38$ $SD = 2.07$; $W = 27.000$ $p = 0.017^{*}$). They also found it easier to adjust their rankings (*ours*: $M = 6.13$ $SD = 1.13$ *vs. baseline*: $M = 3.63$ $SD = 2.33$; $W = 20.000$ $p = 0.029^{*}$) and weights (*ours*: $M = 6.25$ $SD = 1.39$ *vs. baseline*: $M = 3.13$ $SD = 2.36$; $W = 20.000$ $p = 0.029^{*}$) when using our tool. Furthermore, participants found it significantly easier to add new criteria with our tool (*ours*: $M = 6.75$ $SD = 0.71$ *vs. baseline*: $M = 4.13$ $SD = 2.03$; $W = 21.000$ $p = 0.018^{*}$) and to assign and adjust scores (*ours*: $M = 6.13$ $SD = 1.46$ *vs. baseline*: $M = 4.38$ $SD = 1.92$; $W = 28.000$ $p = 0.010^{**}$).

9 Study 2: Case Studies

Building on the insights from our previous study, we conducted two follow-up case studies, where new participants were given two more difficult and nuanced ranking tasks (i.e., ranking short films and grading open-ended projects) using our tool. While the initial within-subjects study showed significant results, ranking is a thoughtful process that requires sustained focus, which these two tasks were better suited to test. These follow-up studies aimed to gather richer qualitative feedback through think-aloud protocols and in-depth interviews, addressing the need for a deeper exploration of users' decision-making and reasoning processes that account for the perceived improvements observed in the last study.

9.1 Case Study 1: Ranking Short Films

9.1.1 Methodology. 4 participants (3 female, 1 male; P09—P12) aged 23—28 ($M = 25.5$, $SD = 2.38$) were recruited for the first case study. Each participant was asked to rank a set of 10 one-minute short films. These films were high-performing entries in a real short film competition over various years [12], meaning that they were evaluated and ranked against similar films in a high-stakes competition. This dataset provided rich, subjective content that resisted purely objective evaluation, encouraging participants to construct and apply diverse decision frameworks during the ranking process. By engaging with films that vary in narrative style, visual aesthetics, and emotional impact, participants were compelled to articulate personal criteria and negotiate trade-offs among them. This setting created a natural opportunity to observe how participants use our tool to externalize their reasoning and resolve competing evaluation dimensions.

The study started from obtaining informed consent and collecting demographic information. Then we provided a demonstration of our tool, introduced its key features, and allowed participants 5 minutes to practice the tool. After the practice session, participants

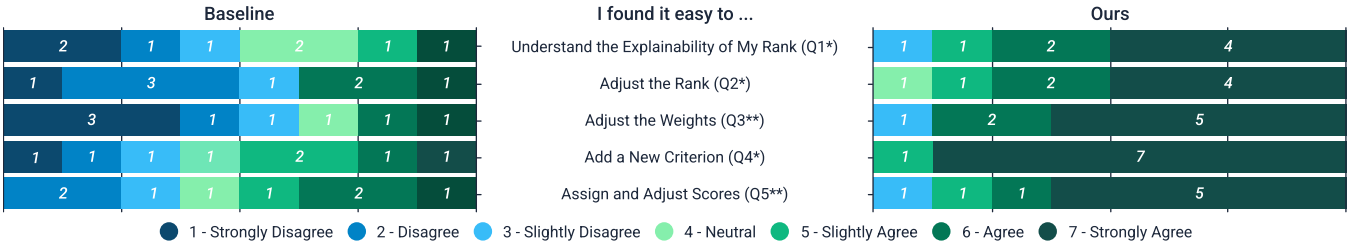

Figure 11: Participants' responses to a 7-point self-defined Likert scale questionnaire, measuring their perceived ease of each actions when creating explainable rankings in each condition.

were tasked with ranking the films in 30 minutes. During the ranking task, participants were instructed to think aloud, verbalizing their thought process and decision-making. Following the task, we conducted an in-depth interview to gather qualitative insights into their experiences, focusing on their decision-making, reasoning, and interaction with the tool. Each session lasted approximately 60 minutes, and participants were compensated with $20 for their time.

9.1.2 Findings. From the first case study, we identified five key findings (KF1—KF5) that illustrate how our tool supported participants' reasoning, reflection, and decision-making throughout the ranking process.

KF1: From Intuition to Structured Reasoning: Participants commonly began the task with intuitive judgments and emotional responses to the content. Then, they used our tool to retrospectively identify and articulate the criteria behind their preferences. This transition from feeling to formalization was often facilitated by the flexiblity of creating and modifying criteria. As P10 noted, her initial rankings were '*purely based on intuition,*" but ranking with the tool helped her "*layer in reasoning*" and retroactively construct criteria. Similarly, P09 commented, "*based on my intuition, I made up the list without any criteria. It's just like my first thoughts on everything. And then I tried to reason like why I make that list.*" Notably, previous multi-criteria DMTs offer minimal support for this mode of interaction.

KF2: Automatic Weighting for Preference Discovery: The automatic weight estimation feature played a critical role in helping participants discover and articulate their implicit preferences. By algorithmically inferring weights from their initial rankings, the tool revealed preference patterns participants hadn't consciously recognized. P10 shared her feelings: "*I didn't think of weighing them, like, really differently before I clicked on it. And I didn't also actively think about, like, meaning as... less important than aesthetics whatsoever.*" Similarly, P09 described using "*the automatic, like, generate weights, makes the number more clear, but the tendency is similar, I think the same as, what I, like, subjectively think about.*" This feature effectively translated participants' intuitive rankings into explicit numerical weights, making implicit preferences visible, and providing a foundation for more refined decision-making as they progressed through the task.

KF3: Explainability Visualization as a Reflective Mirror: Participants found the Rank Comparison Panel, including the slope chart and th conflict highlights, especially helpful in prompting critical reflection on explainability. These visual tools acted as a mirror, surfacing tensions between their stated criteria and actual rankings. P10 described how the visualization created moments of realization: "*When I was playing around... I saw some of the conflicts and then I started to reflect like, oh, maybe I do think this is more important... For example, if I change this, I can see like very visually how these conflicts happen. So, you know, like when we're making decisions, there's a lot of like dilemmas or like debating process. Like, oh, I'm not so sure, but maybe next second I will forget what I was not so sure about. But this tool is like externalizing not only the metrics but also the conflicts as well.*"

KF4: Pairwise Comparison for Resolving Ranking Conflicts: Participants used the Choice Comparison Panel as a critical tool for resolving ranking conflicts. For example, P12 emphasized that it "*definitely helps a lot for me to sort out individual conflicts.*" P10 used the feature "*two to three times*" to reflect on whether they had "*scored them incorrectly or if I should change the weight.*" P09 described how this feature helped them revisit difficult decisions: "*I just looked through those two videos again and I feel like oh yes maybe I should switch the order of those two.*" Beyond local ordering decisions, pairwise comparisons triggered reflection on criteria weightings. When comparing specific items, P12 realized "*this effort spent, it's taking too much weight in calculating which is the best film. So I turn it down a little bit.*" This suggests pairwise comparison prompts users to reconsider not just individual rankings but their entire evaluation framework.

KF5: Tension between Quantification and Subjective Experience: Despite the benefits of our tool, participants highlighted the inherent challenge of translating subjective feelings into quantitative scores. This tension emerged particularly when dealing with nuanced comparisons. P09 articulated this struggle clearly: "*It's hard to interpret everything in number or math.*" P11 described a similar struggle when forced to translate emotional reactions into structured criteria: "*If you let me to write down something, I can definitely write down a paragraph for each video. But if you let me to quantify my emotions... it's super hard.*" This emotional labor may led to decision fatigue. One possible strategy could be to allow free-form natural language reflection inputs and automatically extract criteria from the text.

9.2 Case Study 2: Grading Open-ended Projects

9.2.1 Methodology. 4 teaching assistants (1 female, 3 male; aged 22—27; P13—P16) from a senior-level Computer Graphics course

at our university participated in the second case study. They all have extensive teaching assistant experience (ranging from 2 to 6 semesters). Participants were asked to rank 10 anonymized submissions from the C2 project mentioned in Section 3 drawn from a previous offering of the course. The assignment required teams to design and render a 3D scene with a custom ray tracer. They were instructed to implement at least two features related to course content, and provided with several examples of what constituted a suitable feature. Students were also encouraged to come up with their own features, and told that creativity would be rewarded if the value of the feature was clear from their submitted image or report. This led to a large variety of features, and, in many cases, the same feature appearing in multiple submissions at very different levels of execution quality. For our case study, we selected 10 projects semi-randomly based on the final scores they received in 2024. Here, we tried to meet three criteria. First, the selected projects should cover a large range of scores. Second, there should be at least one "cluster" of similar or identical scores, to ensure that graders will have to make difficult decisions. And third, we made sure to include an example submission created by the instructor, as this submission is not subject to the same student privacy regulations and can be included in our supplemental material.

For each submission, we provided participants with the artifacts originally used for grading, including a 1–3 page PDF describing the concept, technical approach, and implementation details and the rendered image or video. This dataset, which comes from a real grading scenario, reflects the challenge of grading open-ended creative projects in which artifacts vary widely in technical and aesthetic features and must be judged across multiple, potentially competing dimensions. TAs were encouraged to construct, refine, and weight their own evaluation criteria within the tool, then produce an overall ranking of the 10 projects. All materials were de-identified and presented in randomized order.

The study started from obtaining informed consent, collecting demographic information, and a demonstration of our tool. After 5-min practice session, participants were tasked with ranking the projects using our tool with no time limit. As when using our tool for grading in the actual course (and as dictated by privacy regulations), we disabled the AI feature in this case study. On average, participants spent 36.75 minutes ($SD = 14.31$) in the ranking task. The study was closed with an in-depth interview to gather qualitative insights into their experiences. The whole session lasted approximately 75—90 minutes. Participants were compensated with $40 USD for their time.

9.2.2 Findings. From the second case study, we first conducted Spearman's rank correlation tests to examine the consistency between the rankings produced by participants using our tool and the final scores[4] previously assigned to students in the course. The results showed that all participants' rankings exhibited a very strong positive monotonic relationship with the final ranking (P13: $r_s = 0.99$, $p < 0.001^{***}$; P14: $r_s = 0.92$, $p < 0.001^{***}$; P15: $r_s = 0.83$, $p = 0.003^{**}$; P16: $r_s = 0.89$, $p < 0.001^{**}$). All consistency scores were higher than those achieved with the previous method (Section 3.2), where scores were obtained by averaging two graders ($r_s = 0.74$).

[4]The final scores were determined through several hours of review by the teaching team and can therefore be regarded as the ground truth in this case.

Participants also rated the usefulness of each feature on a scale from 1 to 7. All participants gave the Criteria Panel the highest score of 7. The UIS and the Choice Comparison Panel both received an average score of 6.75 ($SD = 0.50$). The Rank Comparison Panel received an average score of 6.25 ($SD = 0.50$). We then identified four key qualitative findings (KF6—KF9) that further illustrate how these features supported users in ranking open-ended student projects.

KF6: Iterative Decomposition of Vague Criteria into Concrete Sub-Criteria: Participants defined different sets of criteria. P13 and P14 focused on three criteria: report quality, scene aesthetics, and technical features. P15 and P16 began with high-level notions (e.g., "technical," "creativity") and then subdivided them into more detailed facets to compare projects in a more fine-grained way. For example, P15 decomposed creativity into scene layout, scene creativity, and feature creativity, while P16 defined criteria for each technical feature, such as texture mapping, refraction, and acceleration.

The flexibility to add new criteria was one of the favorite features for P13, P14, and P16. For instance, P16 realized that "*the whole technical concept is too ambiguous.*" P16 then proceeded to "*break down the technical into different areas, like, do they implement this feature, and how well did they implement this feature,*" describing the overall approach as "*almost like the divide and conquer algorithm,*" where technical concepts were subdivided into specific features, making comparisons more objective: "*After I break down the technical into different areas... it's much easier for me to compare them relative to that single criteria.*" This finding illustrates how our tool helped participants address challenges related to adaptability in explainable ranking, as described in Section 4.2.

KF7: UIS Reduced Biases and Improved Consistency: The UIS is another favorite feature among participants (P13, P14, P15). It helped participants revisit earlier judgments, surface inconsistencies, and mitigate potential biases in their rankings. The findings here demonstrate how our tool achieved the explainability design goal. As P13 remarked, "*insertion sort can be useful to see if the current rankings agree with my current picture of the rankings.*" P15 highlighted how this process compared favorably to traditional grading, emphasizing that it encouraged more systematic reflection that matters to consistency:

> *"Compared to a traditional grading process, I think you're consistently re-evaluating projects and re-evaluating your own criteria in a way that you wouldn't originally do. But [it] kind of helps remove the bias of, like, I viewed this project first... or you look at something, and you're maybe grading it a little wrong, just based on aesthetics. Then you need to actually explain that anyway after you do the insertion sort."* — P15

They further emphasized how the insertion sort simplified bias correction:

> *"When we were grading for graphics, it's like… there would be some things in completely unexplainable sections of the grade, and we would have to debate for a long time to reorder. And, um, I think with this, it's pretty easy to figure out when you're just overrating something for some biased reason, right?"* — P15

Estimated weights further acted as a consistency check. As P15 noted, "*If the estimated weights didn't fit what I wanted in the weights... then it probably meant that my sorting was not treating the criteria according to the weights that I wanted.*" This process was framed as a matter of fairness in grading: "*if I was turning it into a student... that doesn't make sense, and it's not justifiable.*" This finding illustrates how user insertion sort serves as an ethical guardrail, prompting users to approach optimization with caution.

KF8: Broader Applicability Beyond Grading: All four participants saw potential applications of the tool beyond grading, particularly for complex, multi-criteria decision-making. P13 imagined applying it to graduate school decisions: "*I was using Google Spreadsheets, but you can't really do pairwise comparisons. I think, yeah, for a more rational choice, I think it would be useful.*" P14 suggested everyday scenarios like "*buying a car,*" where "*there's a lot of impact, and a lot of factors to consider.*" P15 mentioned life choices such as apartment hunting or prioritizing hobbies: "*You could rank your apartments based on a bunch of criteria... or even do the same kind of thing for hobbies.*" P16 extended it to institutional settings: "*When the company need to recruit anyone, and the government need to give grants... anything when you need to select people and there are a lot of considerations... it would be very useful.*"

KF8: Observed Workflows of Explainable Ranking: Across both studies, participants converged on a similar two-phase workflow when using our system, local and global assessment, which is consistent with what we discussed in Section 3.2.1. Broadly, they first engaged in local, item-by-item evaluation to construct criteria and scores, and then shifted to a global consistency phase to audit and refine their overall explanation.

Phase 1: Local Assessment. In the first phase, participants focused on each item in isolation; they examined individual projects or options to surface relevant properties and articulate candidate criteria. They (1) reviewed each project or option individually to build an intuitive sense of the set, (2) defined criteria while examining items, often adding or renaming criteria as new patterns emerged, (3) scored each project on each criterion, and (4) assigned initial weights based on their early impressions. During this phase, participants primarily interacted with the Target Rank Panel, the Criteria Panel, the Scoring Panel, and the Weights Panel.

Phase 2: Global Assessment. In the second phase, participants shifted from item-level judgments to set-level consistency checks; they evaluated how well their current explanation accounted for the entire ranking. Once they were satisfied that the criteria and weights roughly captured their preferences, they began using UIS, the Rank Comparison Panel, and the Choice Comparison Panel for auditing and repairing their explanations. Specifically, they (5) run UIS to test consistency, (6) inspect flagged conflicts via pairwise comparisons, and (7) resolve contradictions by adding new criteria, adjusting existing criteria, modifying weights, or reordering ranks, repeating Steps 5-7 until conflicts were resolved.

10 Discussion

10.1 Subjectivity, Nuance & Adaptability

Previous multi-criteria DMTs (e.g., [37]) have achieved scalability at the expense of adaptability by restricting exploration weights, which limits rankings to those that are explainable by construction. However, our user studies echo the observation that adaptiveness becomes more critical as evaluations becomes more subjective and nuanced and important criteria become harder to predict. Our tool achieves adaptability by letting the user make independent edits to each of $\{\preceq_*, \mathbf{X}, \mathbf{w}\}$, and then offering flexible ways to understand and resolve contradictions as they arise. This flexibility is crucial to usage patterns that users found most valuable in our user studies.

10.2 Scaling Subjective Evaluation is Hard, But Well-Designed Tools Can Help

Scaling subjective evaluation without explicitly relying on AI remains fundamentally difficult. At best, our strategy of using UIS reduces $O(n^2)$ effort to $O(n)$. This is a significant improvement, but $O(n)$ cost in human time can often still be substantial. Our experience with grading offers some additional insight here. Anecdotally, our grading tasks typically involve ranking between 50 and 100 open-ended projects. Before our first prototype ranking system, we failed to converge on a ranking after 12 hours, at which point we resorted to assigning coarser grades[5]. In more recent offerings with an earlier prototype of our system, similar ranking tasks have converged in under 5 hours. Even if this cost were to scale perfectly linearly, which would be optimal, scaling to say, thousands of projects, would be impractical without the use of some kind of AI or heuristic. This remains an interesting challenge for future work.

10.3 Pairwise Comparisons Capture Subjectivity

In this study, the Choice Comparison Panel emerged as a crucial tool for resolving ranking conflicts. Participants frequently used it to revisit challenging decisions, reflecting on whether their initial rankings were accurate or if adjustments were needed. As participants compared items side by side, they not only clarified their preferences but also reconsidered the criteria and weightings of criteria that guided their evaluations. This reflective process highlights the value of pairwise comparisons in capturing the subjective nature of rankings, enabling users to make more thoughtful, informed decisions.

10.4 The Number of Criteria Used

There is often a tradeoff between the number of criteria used and the precision with which those criteria need to be evaluated, which is closely related to the diversity of choices being considered. If a relevant feature is unique to a single choice, then a criteria based on that feature effectively takes on a binary value. Creating a large number of such binary features can allow for very precise explanations, but it also complicates those explanations, especially when the relevant features are not directly comparable. In such cases, it is often more practical to group features into less precise, more subjective criteria (e.g., the "technical" vs "creative" criteria in our grading case study).

10.5 Ethical Considerations

Our approach to interactive explainable ranking made ethical guardrails an explicit design goal, which led us to use AI and optimization

[5]It was the pandemic. Compromises were made.

in very modular and optional ways. This has the advantage of letting us adapt the approach to different scenarios depending on different ethical considerations. UIS, in particular, is promising as a way to potentially benefit from the use of certain information or technology while at least partially mitigating some of the risks. However, there are open questions. We have not examined the degree to which biases in the insertion order used for UIS might leak through to users, e.g., through anchoring and adjustment. We also have not explored how a bad-faith user might abuse tools that make explainable ranking easier. We do know that disabling features like weight estimation with SVMs can make hunting for bad-faith explanations harder, but this comes at the cost of an opportunity to build meta-cognitive awareness, and we have not explored how much of a barrier it is to bad-faith use of a tool like ours. Ultimately, we believe that the potential for more intelligent explainable ranking tools to improve fairness and do good significantly outweighs these concerns, but they are still important to study.

10.6 Limitations & Future Work

Our system addresses key challenges in explainable ranking, but it also has limitations that future work could address.

10.6.1 Information Gathering Challenges. As users make decisions, they may need to gather information about each choice, especially when scoring based on criteria they may not be familiar with. Since our tool mainly focuses on the stage where users make decisions on ranking, weights, and criteria, it relies on their ability to gather information effectively. When scaling up, gathering a large amount of information for a huge set of choices can become cumbersome. Future work could explore how to integrate an adaptable and scalable information-gathering process into the tool.

10.6.2 Balancing Subjectivity and Quantification. Another limitation of the tool lies in its reliance on user's ability to translate qualitative, subjective experiences into structured, quantitative data. This tension between personal, emotional responses and numerical scores can impact the decision-making experience, potentially leading to fatigue and frustration when nuanced feelings are involved. Addressing this challenge may require new ways to balance numerical scoring with more flexible, free-form expressions.

10.6.3 Supporting Complex Logical Conditions. In future work, we also plan to extend our interface beyond simple weighted sums to support richer logical conditions over criteria (e.g., conjunctions and disjunctions such as "a project must satisfy both technical rigor and creativity," or "a city is acceptable if it satisfies either affordability or safety"). Our current design can only approximate these patterns with linear weights. Beyond single-item evaluation, many real decisions involve reasoning over combinations of items, such as forming a team where the selected group must collectively satisfy multiple requirements (e.g., ensuring that at least one member has leadership experience and at least two have strong technical skills). Supporting these composite, set-level rules introduces new design challenges. A natural next step is to explore interaction and visualization techniques for building, inspecting, and debugging such composite rules while still keeping explanation–rank resolution understandable and manageable for users.

10.6.4 Deployment and Longitudinal Study. While we computed statistical significance in the with-in subjects study, we do not claim the results to be conclusive due to the small sample size. In future work, we see three complementary directions for more ecologically valid evaluation. First, we are already preparing a semester long deployment with teaching assistants in a large course, and plan to introduce the tool to additional instructors in our department. This will let us study how explainable ranking supports authentic grading workflows over months, including evolving criteria, rubric negotiation, and coordination among TAs and instructors. Second, we aim to deploy the system in everyday decision making where the stakes are higher, such as selecting Ph.D. applicants, choosing award recipients, or allocating limited resources. These settings would allow us to examine how the tool influences perceived fairness, justification practices, and conflict resolution when users genuinely care about the outcomes. Third, we plan within system ablation studies that selectively enable or disable key components, such as the Criteria Panel, Rank Comparison Panel, Choice Comparison Panel, UIS, and optimization/AI features. By combining logs, think aloud protocols, and comparative analyses across ablations, we can better isolate which design elements drive improvements in decision-making, consistency, and user confidence.

11 Conclusion

Our work examines a general class of explainable ranking problem and identifies useful design goals for explainable ranking tools. We explore these goals from several dimensions, analyzing previous DMTs and deriving a new design that helps to address explainability, adaptability, scalability, and ethical guardrails through two key design innovations—Explanation-Rank Resolution and User Insertion Sort. We validate our system through a within-subjects study and two case studies, demonstrating that our tool leads to more consistent, explainable rankings and greater user confidence. We believe that our system and many of the insights presented in this work will have significant impact on the way people use technology to scale important and nuanced decisions involving many choices.

References

[1] Paula Akemi Aoyagui, Kelsey Stemmler, Sharon Ferguson, Young-ho Kim, and Anastasia Kuzminykh. 2025. A Matter of Perspective(s): Contrasting Human and LLM Argumentation in Subjective Decision-Making on Subtle Sexism. arXiv:2502.14052 [cs] doi:10.1145/3706598.3713248

[2] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Studies* 4, 3 (May 2009), 114–123.

[3] Clara-Maria Barth, Jenny Schmid, Ibrahim Al-Hazwani, Madhav Sachdeva, Lena Cibulski, and Jürgen Bernard. 2023. How Applicable Are Attribute-Based Approaches for Human-Centered Ranking Creation? *Computers & Graphics* 114 (Aug. 2023), 45–58. doi:10.1016/j.cag.2023.05.004

[4] Zana Buçinca, Maja Barbara Malaya, and Krzysztof Z. Gajos. 2021. To Trust or to Think: Cognitive Forcing Functions Can Reduce Overreliance on AI in AI-Assisted Decision-Making. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1 (April 2021), 188:1–188:21. doi:10.1145/3449287

[5] Shiye Cao, Anqi Liu, and Chien-Ming Huang. 2024. Designing for Appropriate Reliance: The Roles of AI Uncertainty Presentation, Initial User Decision, and User Demographics in AI-Assisted Decision-Making. *Proc. ACM Hum.-Comput. Interact.* 8, CSCW1 (April 2024), 41:1–41:32. doi:10.1145/3637318

[6] Olivier Chapelle and Zaïd Harchaoui. 2004. A Machine Learning Approach to Conjoint Analysis. In *Advances in Neural Information Processing Systems*, Vol. 17. MIT Press, Cambridge, MA, USA, 257–264. https://proceedings.neurips.cc/paper_files/paper/2004/hash/4bbdcc0e821637155ac4217bdab70d2e-Abstract.html

[7] Christopher Cherniak. 1981. Minimal Rationality. *Mind, New Series* 90, 358 (1981), 161–183.

[8] Lacey J. Davidson. 2022. *Implicit Bias and Decision-Making.* Routledge, New York, NY, USA, 13.

[9] Evanthia Dimara, Anastasia Bezerianos, and Pierre Dragicevic. 2018. Conceptual and Methodological Issues in Evaluating Multidimensional Visualizations for Decision Support. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (Jan. 2018), 749–759. doi:10.1109/TVCG.2017.2745138

[10] Julia Dressel and Hany Farid. 2018. The accuracy, fairness, and limits of predicting recidivism. *Science Advances* 4, 1 (Jan. 2018), eaao5580. doi:10.1126/sciadv.aao5580

[11] Jessica Maria Echterhoff, Matin Yarmand, and Julian McAuley. 2022. AI-Moderated Decision-Making: Capturing and Balancing Anchoring Bias in Sequential Decision Tasks. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22).* Association for Computing Machinery, New York, NY, USA, 1–9. doi:10.1145/3491102.3517443

[12] FILMINUTE. 2025. Filminute.

[13] Samuel Gratzl, Alexander Lex, Nils Gehlenborg, Hanspeter Pfister, and Marc Streit. 2013. LineUp: Visual Analysis of Multi-Attribute Rankings. *IEEE Trans. Visual. Comput. Graphics* 19, 12 (Dec. 2013), 2277–2286. doi:10.1109/TVCG.2013.173

[14] Gaole He, Lucie Kuiper, and Ujwal Gadiraju. 2023. Knowing About Knowing: An Illusion of Human Competence Can Hinder Appropriate Reliance on AI Systems. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23).* Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3544548.3581025

[15] Irving Janis and Leon Mann. 1978. *Decision Counseling: Theory, Research, and Perspectives for a New Professional Role.* Springer US, Boston, MA, 145–170. doi:10.1007/978-1-4684-2487-4_10

[16] Irving L. Janis. 1959. Decisional Conflicts: A Theoretical Analysis. *The Journal of Conflict Resolution* 3, 1 (1959), 6–27.

[17] Irving L. Janis and Leon Mann. 1977. *Decision making: A psychological analysis of conflict, choice, and commitment.* Free Press, New York, NY, US. xx, 488 pages.

[18] Thorsten Joachims. 2002. Optimizing Search Engines Using Clickthrough Data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* ACM, Edmonton Alberta Canada, 133–142. doi:10.1145/775047.775067

[19] Daniel Kahneman. 2012. *Thinking, fast and slow.* Penguin, London.

[20] Keith Kirkpatrick. 2017. It's not the algorithm, it's the data. *Commun. ACM* 60, 2 (Jan. 2017), 21–23. doi:10.1145/3022181

[21] James R. Lewis, Brian S. Utesch, and Deborah E. Maher. 2013. UMUX-LITE: When There's No Time for the SUS. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13).* Association for Computing Machinery, New York, NY, USA, 2099–2102. doi:10.1145/2470654.2481287

[22] Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, Ziang Xiao, and Ming Yin. 2025. From Text to Trust: Empowering AI-Assisted Decision Making with Adaptive LLM-Powered Analysis. arXiv:2502.11919 [cs] doi:10.48550/arXiv.2502.11919

[23] Zhuoran Lu, Dakuo Wang, and Ming Yin. 2024. Does More Advice Help? The Effects of Second Opinions in AI-Assisted Decision Making. *Proc. ACM Hum.-Comput. Interact.* 8, CSCW1 (April 2024), 217:1–217:31. doi:10.1145/3653708

[24] Shuai Ma, Xinru Wang, Ying Lei, Chuhan Shi, Ming Yin, and Xiaojuan Ma. 2024. "Are You Really Sure?" Understanding the Effects of Human Self-Confidence Calibration in AI-Assisted Decision Making. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24).* Association for Computing Machinery, New York, NY, USA, 1–20. doi:10.1145/3613904.3642671

[25] Narges Mahyar, Weichen Liu, Sijia Xiao, Jacob Browne, Ming Yang, and Steven P. Dow. 2017. ConsesnsUs: Visualizing Points of Disagreement for Multi-Criteria Collaborative Decision Making. In *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17 Companion).* Association for Computing Machinery, New York, NY, USA, 17–20. doi:10.1145/3022198.3023269

[26] OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, A. J. Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. GPT-4o System Card. arXiv:2410.21276 [cs] doi:10.48550/arXiv.2410.21276

[27] Snehal Prabhudesai, Leyao Yang, Sumit Asthana, Xun Huan, Q. Vera Liao, and Nikola Banovic. 2023. Understanding Uncertainty: How Lay Decision-Makers Perceive and Interpret Uncertainty in Human-AI Decision Making. In *Proceedings of the 28th International Conference on Intelligent User Interfaces (IUI '23).* Association for Computing Machinery, New York, NY, USA, 379–396. doi:10.1145/3581641.3584033

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs] doi:10.48550/arXiv.2103.00020

[29] Jenny Schmid, Lena Cibulski, Ibrahim Al-Hazwani, and Jürgen Bernard. 2022. RankASco: A Visual Analytics Approach to Leverage Attribute-Based User Preferences for Item Rankings. *EuroVis Workshop on Visual Analytics (EuroVA)* 1, 1 (June 2022), 7–11. doi:10.2312/eurova.20221072

[30] Jakob Schoeffer, Maria De-Arteaga, and Niklas Kühl. 2024. Explanations, Fairness, and Appropriate Reliance in Human-AI Decision-Making. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3613904.3642621

[31] Esther-Mirjam Sent. 2018. Rationality and bounded rationality: you can't have one without the other. *The European Journal of the History of Economic Thought* 25, 6 (Nov. 2018), 1370–1386. doi:10.1080/09672567.2018.1523206

[32] L. L. Thurstone. 1927. A law of comparative judgment. *Psychological Review* 34, 4 (1927), 273–286. doi:10.1037/h0070288

[33] James Tompkin, Kwang In Kim, Hanspeter Pfister, and Christian Theobalt. 2017. Criteria Sliders: Learning Continuous Database Criteria via Interactive Ranking. In *British Machine Vision Conference (BMVC)*. BMVA Press, London, UK, 1—15.

[34] Amos Tversky and Daniel Kahneman. 1971. Belief in the law of small numbers. *Psychological Bulletin* 76, 2 (1971), 105–110. doi:10.1037/h0031322

[35] Amos Tversky and Daniel Kahneman. 1974. Judgment under Uncertainty: Heuristics and Biases: Biases in Judgments Reveal Some Heuristics of Thinking under Uncertainty. *Science* 185, 4157 (Sept. 1974), 1124–1131. doi:10.1126/science.185.4157.1124

[36] Amos Tversky and Daniel Kahneman. 1981. The Framing of Decisions and the Psychology of Choice. *Science* 211, 4481 (Jan. 1981), 453–458. doi:10.1126/science.7455683

[37] Emily Wall, Subhajit Das, Ravish Chawla, Bharath Kalidindi, Eli T. Brown, and Alex Endert. 2018. Podium: Ranking Data Using Mixed-Initiative Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (Jan. 2018), 288–297. doi:10.1109/TVCG.2017.2745078

[38] Xinru Wang. 2024. Human-Centered Evaluation of Explanations in AI-Assisted Decision-Making. In *Companion Proceedings of the 29th International Conference on Intelligent User Interfaces (IUI '24 Companion)*. Association for Computing Machinery, New York, NY, USA, 134–136. doi:10.1145/3640544.3645239

[39] Xinru Wang and Ming Yin. 2022. Effects of Explanations in AI-Assisted Decision Making: Principles and Comparisons. *ACM Trans. Interact. Intell. Syst.* 12, 4 (Nov. 2022), 27:1–27:36. doi:10.1145/3519266

[40] Di Weng, Ran Chen, Zikun Deng, Feiran Wu, Jingmin Chen, and Yingcai Wu. 2019. SRVis: Towards Better Spatial Integration in Ranking Visualization. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan. 2019), 459–469. doi:10.1109/TVCG.2018.2865126

[41] Di Weng, Heming Zhu, Jie Bao, Yu Zheng, and Yingcai Wu. 2018. HomeFinder Revisited: Finding Ideal Homes with Reachability-Centric Multi-Criteria Decision Making. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. doi:10.1145/3173574.3173821

[42] Jian-Bo Yang and M.G. Singh. 1994. An evidential reasoning approach for multiple-attribute decision making with uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics* 24, 1 (Jan. 1994), 1–18. doi:10.1109/21.259681

[43] Yisong Yue and Thorsten Joachims. 2009. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML '09)*. Association for Computing Machinery, New York, NY, USA, 1201–1208. doi:10.1145/1553374.1553527