

A Preference Judgment Interface for Authoritative Assessment

by

Mahsa Seifikar

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2022

© Mahsa Seifikar 2022

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Prof. Charles Clarke, my supervisor, contributed directly to the design and implementation of the preference judgment algorithm presented in Chapter 3.3 including coding and testing. I am the sole author of the remaining parts.

Abstract

For offline evaluation of information retrieval systems, preference judgments have been demonstrated to be a superior alternative to graded or binary relevance judgments. In contrast to graded judgments, where each document is assigned to a pre-defined grade level, with preference judgments, assessors judge a pair of items presented side by side, indicating which is better. Unfortunately, preference judgments may require a larger number of judgments, even under an assumption of transitivity. Until recently they also lacked well-established evaluation measures. Previous studies have explored various evaluation measures and proposed different approaches to address the perceived shortcomings of preference judgments. These studies focused on crowdsourced preference judgments, where assessors may lack the training and time to make careful judgments. They did not consider the case where assessors have been trained and provided with the time to carefully consider differences between items. We review the literature in terms of algorithms and strategies for extracting preference judgment, evaluation metrics, interface design, and use of crowdsourcing. In this thesis, we design and build a new framework for preference judgment called JUDGO, with various components designed for expert reviewers and researchers. We also suggested a new heap-like preference judgment algorithm that assumes transitivity and tolerates ties. With the help of our framework, NIST assessors found the top-10 best items of each 38 topics for TREC 2022 Health Misinformation Track, with more than 2,200 judgments collected. Our analysis shows that assessors frequently use the search box feature, which enables them to highlight their own keywords in documents, but they are less interested in highlighting documents with the mouse. As a result of additional feedback, we make some modifications to the initially proposed algorithm method and highlighting features.

Acknowledgements

I would like to express my sincere appreciation to my supervisor, Prof. Charles L.A. Clake, for his constant support and guidance throughout my graduate studies. I would like to thank Prof. Mark D. Smucker for his insightful feedback, especially during system design and implementation.

Last but not least, I want to thank my mother and sister for their eternal love, support and encouragement. Without them, I would not have lived my incredible life this far. I want to thank everyone who has made a significant impact on my life.

Table of Contents

Author's Declaration	ii
Statement of Contributions	iii
Abstract	iv
Acknowledgements	v
List of Figures	viii
List of Tables	x
1 Introduction	1
2 Background	5
2.1 Binary Judgment	5
2.2 Graded Judgment	6
2.3 Preference Judgment	7
2.3.1 Strict vs Weak Preference	8
2.3.2 Transitivity	9
2.3.3 Evaluation Measures	10
2.3.4 Algorithms and Strategies	11
2.3.5 Interfaces and frameworks	13

3	Preference Judgment Interface	15
3.1	Design Considerations	15
3.2	System Overview	16
3.2.1	User Interface Side	18
3.2.2	Administrative Panel Side	22
3.3	Preference Judgment Algorithm	23
3.3.1	Number of Judgment Estimation	27
4	Experiments on TREC 2022 Health Misinformation Track	29
4.1	TREC 2022 Health Misinformation track	29
4.2	Dataset	30
4.3	Run-time Settings	32
4.4	Analysis	35
4.5	Feedback and Discussion	38
5	The System Adjustments	41
5.1	Algorithm Improvements	41
6	Conclusion	45
	References	47

List of Figures

2.1	An overview of different types of relevance judgment	7
2.2	Example of three different interfaces designed by A)[42] B)[61] C)[10]. . .	13
3.1	An overview of the JUDGO interface.	17
3.2	Detailed view of the home page.	18
3.3	Detailed view of judgment page components.	19
3.4	An example of overlap between the mouse-highlighted sentence in yellow and the search box-highlighted keywords	22
3.5	An example of the list of Pref object from a pool with five items: A, B, C, D, and E	23
3.6	An example of the list of Pref objects with five items and two sequences of judgment, such as $(B > A)$ and $(C > D)$	24
3.7	An example of the list of Pref objects. with five items and four sequences of judgment, such that $(B > A)$, $(C > D)$, $(B > E)$ and $(C > B)$	26
3.8	An example of a list of Pref objects when the first round of the algorithm is completed.	28
4.1	The total number of judgment completed in a time frame less than 30 minutes.	36
5.1	An example of two versions of the preference judgment algorithms after first judgment is done: the left-hand side shows the new version, and the right-hand side presents the previous one.	43
5.2	An example of two versions of the preference judgment algorithms after the second judgment is done: the left-hand side shows the new version, and the right-hand side presents the previous one.	43

5.3 An example of two steps in the new version of the preference judgment algorithm, such that $(D > B)$ and $(E > D)$ 44

List of Tables

1.1	Three of documents assigned “Very Useful” by assessors during the TREC 2022 Health Misinformation Track for the question: <i>Can chewing gum help lose weight?</i> with the answer “NO”. Documents have been truncated. . . .	2
2.1	Judgment Relevance Scale description for the TREC 2019 Deep Learning Task	6
2.2	A summary of the prior research work on preference judgment.	14
3.1	List of features in judgment page.	21
4.2	Information about topics that is provided to the JUDGO system.	31
4.1	Sample of topics in TREC 2022 Health Misinformation Task.	33
4.3	Example of one document in TREC 2022 Health Misinformation Task. . .	34
4.4	Result of document ranking for a topic ID 170 which is about Fish Oil. . .	37
4.5	Result of document ranking for a topic ID 163 about Fruit Juice.	39
4.6	Result of document ranking for a topic ID 165 which is about SIT UP. . .	40

Chapter 1

Introduction

Offline evaluation is a widely used approach for measuring the performance of information retrieval systems, including search engines, recommendation systems, and question answering systems[9, 31, 17]. These evaluations typically rely on a “gold standard” of the ideal document ranking for a given set of queries, with relevance judgments being the most commonly used standard[52]. Relevance is not only determined by a document’s topical similarity to a query, but also by other factors such as the authority, quality, and reliability of the document in relation to the user’s information needs[54, 11].

In the traditional approach to collecting relevance judgments, known as binary judgments, a topic and a document are presented to an assessor, who must determine whether or not the document is relevant to the given topic [48]. The binary judgments of irrelevant/relevant have previously been generalized to three or more graded values that are more discriminatory, such as irrelevant/relevant/extremely relevant [56, 33, 7]. However, both binary and graded judgments are independent. Therefore assessors should assign a grade to a document independent of any other document in the collection[54].

Preference judgment has been demonstrated as an excellent alternative to graded judgment [11, 31, 38, 21, 49, 59]. In this approach, the evaluators are presented with two separate documents side-by-side from the collection, and they must determine which one of the documents is more relevant to the topic presented to them [48]. Carterette et al. [11] have demonstrated that assessors make relative judgments more easily, quickly, and reliably than graded judgments. They also indicated that there is a higher level of agreement between assessors for all pairs of judgments and better quality of judgment in preference judgment. Preference judgments are also capable of capturing other factors in addition to those that can be derived from absolute judgments [17].

Table 1.1: Three of documents assigned “Very Useful” by assessors during the TREC 2022 Health Misinformation Track for the question: *Can chewing gum help lose weight?* with the answer “NO”. Documents have been truncated.

en.noclean.c4-train.05939-of-07168.45060: Disappointing news for dieters who chew gum to aid weight loss and the entire REDBOOK staff: A new study in the journal Eating Behaviors researched the effects of chewing gum on food intake and found that it had no effect on the amount of calories consumed (and actually increased meal size in some cases), it also may lead to making poorer food choices...

en.noclean.c4-train.05398-of-07168.95043: The participants’ records showed that regardless of whether they chewed gum or not, there was no difference in the amount of food they ate later in the day. Furthermore, their appetite levels at various points throughout the day were basically the same no matter when (or if) they chewed gum after lunch. I have patients say all the time that they think that gum chewing helps them with their diet. This research doesn’t bear that out...

en.noclean.c4-train.06282-of-07168.45677 : Another common gum myth is that sugar-free gum can help you lose weight. Although it is preferable to choose sugar-free gum over the extra-sweet variety, no studies have shown that sugar-free gum will help you lose weight. If you pop a piece of gum in your mouth after dinner to avoid dessert, it could help you avoid eating a few extra calories every day...

Table 1.1 presents three documents graded as “Very Useful” by the assessors during TREC 2022 Health Misinformation Track ¹ for the question of *Can chewing gum help lose weight?*. As shown, all three documents clearly state that chewing gum has no impact on weight loss; that’s why they were all given the same level at the graded judgment step. With less burden on the assessors, preference judgments enable us to differentiate the documents even more by comparing them side by side. Our experiments indicate that assessors preferred the third document the most and the second document over the first one.

Preference judgments, however, suffer from two significant limitations: first, in comparison to absolute judgments, they are more labor-intensive and demand more effort. A collection of n documents requires $O(n^2)$ pairs of preference judgments. Previous research [11, 49, 12] assumed transitivity, which means that if $d1$ is preferred over $d2$ and $d2$ is preferred over $d3$, then $d1$ should be preferred over $d3$. Preference judgments require $O(n \log n)$ judgments to generate a ranking order even under the transitivity assumption, whereas absolute judgments only require $O(n)$ judgments. There are several papers [21, 46, 42] that try to solve this problem by focusing on determining the ranking order of the top k items. The second drawback is the lack of universally established evaluation measures for preference judgments. Sakai and Zeng [52] established a wide range of evaluation metrics based on [9, 11] works. Over the past few years, Clarke and his research teams [21, 20, 19] have also suggested an evaluation criterion called compatibility.

The majority of earlier studies have concentrated on evaluation and effective strategies to reduce the number of judgments. In this thesis, we design and develop a unified framework called JUDGO for preference judgment that is advantageous to both researchers and assessors. For assessors, we include features such as a search box, the ability to highlight documents, font change buttons, etc., to assist them to read and judge more quickly. In addition, we develop progress bars to provide an approximation of how far they have progressed. We provide three different components for researchers to manage their assessors and monitor the quality of judgment, including quality control, import/export, and task assignment. We also suggest a novel method based on a heap-like data structure that focuses on finding the top- k items for each topic for preference judgment.

We examined our proposed framework on the TREC 2022 Health Misinformation Track from September to October 2022. The assessors, are employed by the National Institute of Standards and Technology (NIST), retrieved the top 10 documents for each topic and ranked them according to their preferences with the help of our tool. We further analyze the assessors’ behavior and to what extent they take advantage of the designed features.

¹<https://trec-health-misinfo.github.io/>

The main contributions of this thesis can be summarised as follows:

- Designed and developed a new preference judgment framework that can meet the needs of both researcher and expert assessors and crowdsource workers.
- Proposed and implemented a novel heap-like preference judgment algorithm that can tolerate transitivity and ties.
- Deployed this framework for the TREC 2022 Health Misinformation track on Heroku.
- Collected more than 2,200 pairs of judgments from expert users and collected the top-10 best documents for 38 topics of the TREC 2022 Health Misinformation track.
- A detailed analysis of user behavior and demonstrated how frequently they use specific components of the tool.

This thesis is organized as follows: In Chapter 2, we first introduce binary judgment, graded judgment, and preference judgment and explain their differences and limitations, then discuss preference judgment in terms of tie, transitivity, evaluation measures, algorithms, and interfaces. Before discussing the suggested preference judgment algorithm, we present a high-level overview of the framework in Chapter 3 and elaborate on each designed component's purpose and functionality. In Chapter 4, we demonstrate how we utilize the suggested framework in the TREC 2022 Health Misinformation track, show our study of user behaviour judgments, and discuss the feedback. The system improvements we adapt based on the lessons we learned from TREC are then discussed in Chapter 5. The final chapter summarizes the key findings of the thesis and discusses potential directions for future works.

Chapter 2

Background

In this chapter, we discuss binary, graded, and preference judgments since they are the three main varieties of relevance assessments that have been employed in the majority of previous studies so far for offline evaluation. Afterward, we discuss the distinctions between strict and weak judgment, along with how the use of transitivity assumptions reduces the total number of document pairs. Following that, we elaborate on various evaluation metrics, algorithms, and designed interfaces that have been suggested for preference judgments. At the end of this chapter, we present a table that provides a comprehensive review of the studies that were conducted on preference judgments.

2.1 Binary Judgment

In the information retrieval research area, binary judgment is a traditional method for relevance judgment. In this approach, documents are either relevant to the presented information need or not, so it can be considered a classification problem [19]. If a set of documents are labelled as relevant, there is no other difference among them [5]. Based on binary relevant judgment, a number of different evaluation metrics have been suggested, such as precision, recall, average precision (AP), and so on [62]. Average precision was widely used as an evaluation metric for many experiments during the early years of Text REtrieval Conference (TREC).[56]. However, there is one main criticism of this approach, namely, that millions of items are in some way relevant to the query.

2.2 Graded Judgment

The graded judgment addresses the limitation of the binary approach by considering relevance as a multi-value scale concept. Each item is assessed separately and assigned a scale based on a set of levels of relevance [31]. Based on this approach, normalized discounted cumulative gain (nDCG)[32], expected reciprocal rank (ERR)[15] and ranked-biased precision (RBP)[45] have been introduced to evaluate the performance of the ranker algorithms. Recently, nDCG has been used as an evaluation metric in both industries and research, for example, TREC employed this metric, particularly for tracks that work on web collections [19, 14, 43, 26]. However, it is challenging to extend this metric to take into consideration additional aspects besides relevance [19].

Graded relevance judgment has three primary drawbacks despite the fact that it has been widely employed by researchers and addresses some significant problems with the previous approach for a long time. The first and most important difficulty is the lack of universally accepted guidance on how to design a graded scale, and most of the previous studies defined different numbers of scale point with varying interpretations [62, 19].

For instance, a five-point scale relevance set (“Fully meets”, “Highly meets”, “Moderately meets”, “Slightly meets”, “Fails to meet”) was employed for the conversational search method in the TREC 2019 Conversational Assistance track (CAst) [24]. While in the TREC 2019 Deep Learning track four-point scale of judgment is adopted (“Perfectly relevant”, “Highly relevant”, “Related”, “Irrelevant”) [23]; The table 2.1 provides the definition of each grade.

Table 2.1: Judgment Relevance Scale description for the TREC 2019 Deep Learning Task

Scale	Description
Perfectly relevant	The passage should appear in the first few results since it perfectly answers the query.
Highly relevant	The paragraph contains some answers which are ambiguous.
Related	The passage does not provide an answer, yet it seems to be relevant to the query.
Irrelevant	The passage is unrelated to the query.

The second issue is that assessors have to take more time when deciding on a grade for each document, particularly when the number of grades is increased and descriptions

of each grade are not clear [11, 24]. Besides the time and effort, huge disagreements have been reported between different judges in previous works [11, 5].

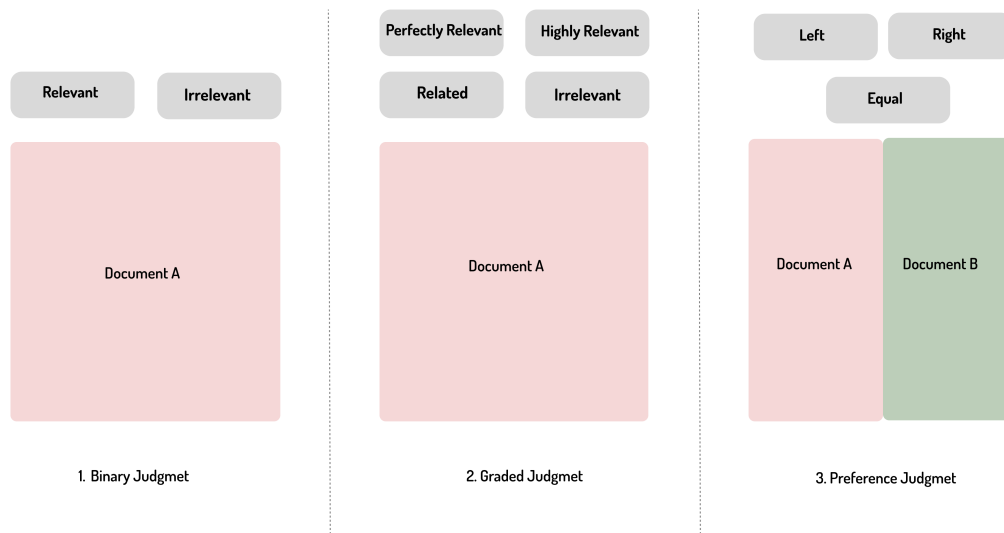


Figure 2.1: An overview of different types of relevance judgment

2.3 Preference Judgment

In the preference judgment approach, the concept of relevance can be understood as the idea that item d_1 is more or less relevant than d_2 or, in some cases, d_1 and d_2 are relevant in the same way [5]. This is different from graded judgment, in which the assessor must judge the relevance of each document in isolation, independent of the other documents in the topic or query before determining a level of relevance [11, 19]. As shown in figure 2.1, in each step a pair (d_1, d_2) will be shown for a topic, and assessors are required to specify their preference after carefully reading both of the items in the pair.

Preferences can be implicitly extracted from an existing relevance score or other signals such as a click, in addition to direct human judgments [34, 63, 36, 25, 6, 55]. For instance, if a user clicks on a document, it can be deduced that the clicked document is the one that the user prefers the most out of all the documents that are ranked higher [52, 35]. It has been demonstrated that preference judgment may be completed more rapidly than graded

judgment, and there is a higher level of agreement among evaluators[11]. Depending on the context, preferences can consider factors impractical to extract from graded judgments; for instance, a closer market in a grocery search, cheaper items in e-commerce and the most recently updated report in a news search may be preferred [17].

According to Kim et al. [39], preference judgment can take into account a variety of dimensions in addition to topical relevance including authority, diversity, caption quality, and freshness. In one of the earliest studies in preference judgment, in 1990, Rorvig [49] demonstrated the importance of preferences by employing a mathematical notion known as “simple scalability” to show that it is essential to use preference to find items that are highly relevant. In addition, he mentioned that in comparison to the graded judgment, the preference judgment might need more effort to evaluate all of the items in the retrieved set.

Carterette et al. [10] created one of the earliest test collections for preference judgment, and they demonstrated, based on an experiment on 50 topics extracted from TREC 2003 Web Track, that preference judgment outperforms graded judgment for every evaluation measure, although the difference was not particularly significant.

However, it has two significant drawbacks. The first is that, even in the case of transitivity, as we further discuss in this section, preference judgment requires more effort and cost on the assessor’s part. Given a collection of n documents, the exact number of pairs of documents is $\binom{n}{2}$ [47]. In graded judgment, it takes $O(n)$ time to assign a level of relevance to all documents, whereas $O(n \log n)$ time is required for preference judgment even under the assumption of transitivity [11]. The second is that these preference judgments do not have as well-established an effective measurement as do traditional relevance judgments [11]. Nevertheless, recent studies have suggested a variety of potential evaluation methods to address these shortcomings.

The most important research studies on preference judgments are shown in Table 2.2, and for each paper, six important aspects are analyzed, including consideration of ties, assumption of transitivity, whether they introduced a novel evaluation metric and strategy, whether they designed a new interface, and whether they used crowdsourcing.

2.3.1 Strict vs Weak Preference

The primary assumption behind the preference judgment is that users should determine whether or not one document is preferred to the other one[62]. Given a finite set of documents D and $d1, d2 \in D$, the assessors are required to indicate the relationship between two documents, which cannot be unknown [17].

In strict preference for documents $d1$ and $d2$, assessors should make a binary decision on " $>$ " relation, so that there are two options; $d1$ is preferred over $d2$ ($d1 > d2$) or vice versa ($d2 > d1$). In the absence of strict judgment where ($not(d2 > d1)$)and($not(d1 > d2)$), the assessor can have the weak preference option ($d1 \sim d2$) [62], which means two documents can be equally relevant or equally irrelevant to the information need [31]. Generally, there are three relations between documents in weak preference, including "better than", "worse than", and "tied with" [30].

In earlier research, different studies relied on a variety of configurations for employing these two types of preferences. There are several works [49, 47, 27] that are conducted with strict preference assumptions for preference assessment. Carterette et al. [11] created a new framework for relevance judgment based on strict preference while also investigating weak preference. In more recent investigations [54, 64, 61, 60], ties have been taken into consideration. For example, Kazai et al. [38] studied the inter-assessors agreement as well as the link between agreement and user satisfaction by employing a set of weak judgments from both crowd workers and editorial judges.

2.3.2 Transitivity

The study of transitivity is one of the most essential parts of preference judgment since it directly impacts the number of judgments and time consumption [11]. Given transitivity, it follows that if document $d1$ is preferred to $d2$, ($d1 > d2$), and $d2$ is also preferred to $d3$, ($d2 > d3$), then ($d1 > d2 > d3$) [12]. Since evaluators do not have to judge each and every pair of documents, this results in a significant reduction in the total number of judgments that need to be made from $O(n^2)$ to $O(n \log n)$ [11].

Using studies covering fifty different topics and the assistance of an undergraduate business student, Roving [49] established for the first time that preference judgment is transitive. In support of a previous study, Carterette et al. [11] discovered that 99% of triplet documents, which were collected from retrieved web pages by Google, Yahoo!, and Microsoft Live search engines, have transitivity holds, which means assessors were consistent with their judgment. They also demonstrated that by employing an early stopping rule, assessors are not required to judge all pairs of documents, resulting in a significant reduction in the total number of judgments, which averaged $O(n)$. In addition, Chandar et al.[12] reported 96% of triplet documents are transitive on data created by Allan et al.[1].

In addition to using a small number of experts, many researchers have investigated the possibility of using crowdsourcing as a way of making relevance judgments [2, 4, 41, 44]. Using TREC Web 2013 and 2014 Tracks, Hui et al. [31] investigated transitivity on relevance

judgments collected from crowdsourcing. They found a substantial difference between strict preference and weak preference: on average, 96% of strict preference judgments hold transitivity, compared to 75% of weak preference judgments.

Yang et al. [61] reported that the use of crowdsourcing produced the same outcomes as the utilization of professional reviewers. They created a collection in order to investigate crowdsourcing with the objective of reducing ties through the aggregation of preference, binary, and ratio assessments.

2.3.3 Evaluation Measures

Frei and Schäuble [27] suggested a new evaluation method for document preference based on the precision of preference for each topic. Their statistical approach indicated which two ranking systems provide more useful results. Furthermore, using the weak ordering of documents, Yao [62] proposed a novel evaluation metric for preference judgment based on a distance function between the users' preferred ranking and the ranking that the system recommended.

In a series of research studies[9, 11], Carterette with other researchers introduced four evaluation metrics for preference judgment including precision of preferences (*ppref*), recall of preferences (*rpref*), weighted precision of preference (*wpref*) and average precision of preferences (*APpref*). Additionally, they demonstrated that there is a high correlation between introduced preference measures and absolute-based measures, and they are, on average, stable.

On the basis of their prior work [12], which focused on the ranking of novel documents, Chandar and Carterette [13] proposed a whole new evaluation measure for the novelty and diversity preference assessment. The newly developed evaluation method is capable of dealing with disagreements in preferences as well as numerous judgments. Bashir et al. [5] proposed a new method for extracting document relevance scores from pair of preferences Elo rating system which is a method for calculating scores in two players games.

Sakai and Zeng [52] proposed a variety of new evaluation measures (27 in total) called Pref measures and Δ -measures for preference judgments. They also reported that they agreed with the user's perception of search engine result pages (SERPs) and they did not need the transitivity assumption. The Pref measures are based on Carterette et al. [11] and Carterette and Bennett [9], which directly employed the pair of judgments. In contrast, Δ -measures, which is based on their previous works[51, 50] used traditional graded measure by converting preferences to the level of values. In addition, they released a dataset of preference judgments that included more than 100,000 document preferences.

Clarke et al. [20] proposed a new evaluation measure called “compatibility,” which measures the maximum similarity between perfect ranking and system ranking. To calculate compatibility, they employed Rank Biased Overlap (RBO) [57]. Since this approach focused on partial preferences of top-k items, it allowed researchers to combine both graded judgment and preference judgment without any changes. In their next paper [20], they demonstrated compatibility might be broadened to include criteria other than significance.

Based on their three previous studies [20, 19, 21], Clarke et al. [17] also defined and validated an assessment measure called Preference Graph Compatibility (PGC). This measure is based on an acyclic graph that computes the similarity between a directed multi-graph of preferences and the actual system ranking. The primary benefit of PGC in comparison to earlier methods of assessment is that it can be applied to any kind of preferences multigraph, regardless of whether the preferences were generated by graded or side-by-side judgment.

Another approach to assess the ranking system besides counting correctly ranked items is to determine whether they can rank the most well-known relevant items at the top of the ranking [59, 52, 61]. In more recent studies, Arabzadeh et al. [3] used the crowdsourcing benefits for side-by-side preference judgments. After re-evaluating the top retrieved items by neural rankers on the MS MARCO, the researchers discovered that the performance of modern neural ranker systems like BERT surpassed a hypothetically perfect ranker for this dataset.

2.3.4 Algorithms and Strategies

The majority of the earlier studies concentrated on the assessment measures for preference judgment; however, the preference process itself received relatively little attention [60]. According to what was mentioned before, if there are n items in a document collection for a specific query, then there are precise $\binom{n}{2}$ pairs of documents to evaluate. In order to reduce the number of documents pairs, Carterette et al. [11] utilized the assumption of transitivity, the “bad” button for judgments and early stopping rules, and they suggested utilizing a sorting algorithm so that a complete preference judgment could be done without any information loss.

Song et al. [54] suggested a new strategy called “Select-the-Best-Ones” (SBO), which is faster than the sorting sort strategy. In SBO, assessors selected the best ones out of a bunch of documents after which they are required to choose one best item repeatedly. Radinsky and Ailon [47] proposed an active learning approach with the goal of determining the rank of top-k; however, they did not provide any other information on this strategy. In

another study, Sanderson et al.[53] simply presented the pairs of documents with a slight difference to the judges.

Niu et al. [46] proposed top-k learning to rank strategy, and to identify the top-k item, they employed a modified heap sort. Additionally, they introduced FocusedRank, a new ranking method. Busa et al. [8] proposed a preference-based racing algorithm to find the top-k items and tested it on sports data. Chen et al.[16] introduced a new approach to combining pairwise comparisons, which are generated through crowdsourcing, to create a gold-standard global ranking. Utilizing a Bayesian framework, they formalized this problem as an active learning strategy.

Bashir et al. [5] calculated the relevance score for all documents in the collection using BM25, then considered all combinations of the six top documents. Hassan Awadallah and Zitouni [29] unitized a machine learning classifier to predict user preference automatically, which resulted in the reduction of human effort. Kallori et al. [37] proposed a new active learning approach to extract pairwise judgment procedures for recommendation systems through mobile experiments.

Based on Carterette et al. [11] study, Clarke et al. [21] proposed a new tournament-like approach focusing on finding the top item from the pool while minimizing the judgment efforts. They designed the process as a single-elimination tournament, but they did not provide the details of the algorithm. To reduce the total number of judgments, they initiated a graded assessment process before the preference judgment process. They employed crowdsourcing for the TREC 2019 Conversational Assistance Track [24] in order to test their suggested process utilizing their newly developed evaluation measure[20] mentioned in previous section 2.3.4.

Yan et al. [60] proposed a new preference judgment process with the aim of minimizing the number of judgments and tolerating ties while finding the best items. They mapped this problem to the dueling bandits that were previously used for evaluation purposes in the IR research area, and they examined various candidate algorithms both from machine learning and previous studies. Through a series of comparison processes, the dueling bandits attempt to determine which of the K arms is the best[28]. After running simulations on the selected method, they determined that the improved version of Clarke’s approach [21] was the most promising candidate. Using crowdsourcing based on the algorithm, they collected more than 10,000 preference-judgment pairs for the TREC 2021 Deep Learning Track [22].

2.3.5 Interfaces and frameworks

The majority of the earlier research designed and introduced unique judgment interfaces in order to conduct their own individual experiments [54, 21, 38]. Carterette et al. [11] developed a user interface that displays documents as the online content of their respective websites as well as their URLs. Additionally, they highlighted query terms to find relevant content faster. In their subsequent studies [10], they also displayed the assessor’s progress as a number for each query. Besides features in the previous study, Chandar and Carterette [12] enhanced their designed preference judgment framework with a progress bar and topic description. They concluded from their experiments that assessors prefer shorter documents with fewer highlighted terms.

Yang et al. [61] designed a simple system that considers both relative and absolute relevance simultaneously. In a more recent paper, Kuhlman et al. [40] designed an interactive complex framework to collect preference judgments. In a more recent year, Li et al. [42] proposed a new framework for preference judgment to find the top-k items, with the goal of securing the quality of crowdsourced judgments and minimizing the total cost. As shown in figure 2.2-A, they designed a budget monitoring panel and suggested the idea of a sliding bar that would allow the assessors to weigh their preference between two items.

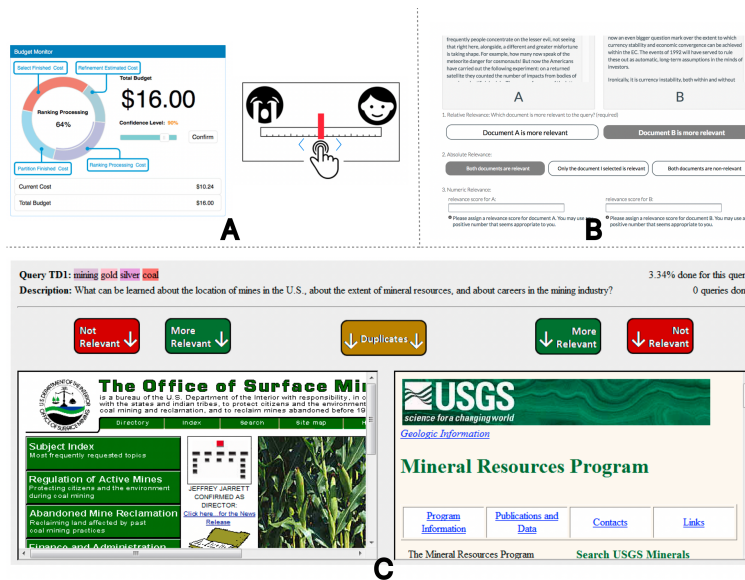


Figure 2.2: Example of three different interfaces designed by A) [42] B) [61] C) [10].

Table 2.2: A summary of the prior research work on preference judgment.

Paper	Ties	Transitivity	Metric	Strategy	Interface	Crowdsource
Bashir et al.[5]	-	-	✓	✓	-	✓
Busa et al.[8]	-	-	-	✓	-	-
Carterette et al.[10]	✓	✓	-	-	✓	-
Carterette et al.[11]	-	✓	✓	✓	✓	-
Chandar et al.[12]	✓	✓	-	-	✓	✓
Chandar et al.[13]	-	-	✓	-	-	-
Chen et al.[16]	-	-	-	-	✓	✓
Clarke et al.[20]	✓	-	✓	-	-	-
Clarke et al.[17]	-	-	✓	-	-	-
Clarke et al.[21]	-	✓	-	✓	✓	✓
Frei and Schäuble[27]	-	-	✓	-	-	-
Hassan et al.[29]	-	-	-	✓	✓	-
Hui and Berberich[30]	✓	✓	-	-	-	-
Kazai et al.[38]	✓	-	-	-	✓	✓
Kim et al.[39]	✓	-	-	✓	-	-
Li et al.[42]	✓	✓	-	-	✓	✓
Niu et al.[46]	✓	✓	✓	✓	-	-
Roitero et al.[48]	✓	-	✓	-	-	✓
Rorving[49]	-	✓	-	-	-	-
Sakai and Zeng[52]	✓	-	-	✓	-	-
Sanderson et al.[53]	✓	-	-	-	✓	✓
Song et al.[54]	✓	-	-	✓	✓	-
Xie et al.[59]	✓	✓	✓	-	-	-
Yang et al.[61]	✓	-	-	-	✓	✓
Yao[62]	✓	✓	✓	-	-	-
Zho and Carterette[64]	✓	-	-	-	-	✓

Chapter 3

Preference Judgment Interface

In this chapter, we discuss the algorithm, design and implementation of the current preference judgment system. We explain the considerations involved in designing the system. Then, we provide a general overview of the system and then discuss the crucial system components and elaborate on their purpose and functionality. Next, we explain the fundamental algorithm for preference judgment behind the system. Finally, depending on the number of documents and the top result, we examine how we estimate the number of total judgments.

3.1 Design Considerations

Based on the previous studies in our research group and several discussions in the JUDGO open-source repository, we determined the following design considerations for the presented preference judgment framework.

D1: Provide an integrated interface to accelerate clear reading and making decisions: Documents might be long or short, and assessors may or may not be experts in the areas they evaluate. To speed up identifying the relevant parts of documents, previous studies[12, 11] highlighted the query terms in the same colour in both documents, but they didn't allow users to enter their own keywords. To make the main keywords distinguishable in both documents, we consider a search box component where users can enter a word or phrase and see them immediately in different colours. Furthermore, none of the earlier studies considered the possibility that an assessor could view a document multiple times during a preference judgment. We believe this issue could be addressed by adding a feature

that would allow users to highlight particular sentences in documents, thereby making it easier for them to identify the key details and accelerate judgment the next time they look back at the same document.

D2: Leverage various pieces of information to enhance topic and document understanding: In earlier research, it was assumed that adding topic explanations or links to each document’s website would help assessors better understand the themes and documents, so we kept the same assumption in our design.

D3: Enable assessors to change their previous decisions: Assessors may wish to revise their initial decisions as they proceed through the judging process. In the present interface, we devise a feature for this demand because, to the best of our knowledge, none of the prior interfaces enabled assessors to accomplish this.

D4: Provide support for a variety of screen sizes: On any sort of device, including a laptop, mobile phone, tablet, etc., assessors should be able to read and see pairs of documents clearly. We devise two features: the first one offers the assessor the ability to change the font size of both documents; the second one is a dragbar that allows the assessor to adjust the space available between the documents on the left and the right.

3.2 System Overview

Figure 3.1 depicts the architecture of the JUDGO system, which consists of three main modules: user interface, backend, and administrative panel. This system is intended for two types of users: researchers who seek to obtain a ranking order for their documents based on the suggested preference judgment algorithm and reviewers who are responsible for reading several pairs of documents and choosing the one that they believe is more relevant to the provided topic. Researchers utilize the admin panel, whereas reviewers interact with the user interface component.

The database and log files are directly accessible from the backend and administrator panel. The designed database for this system has five main entities, including *user*, *topic*, *document*, *task* and *judgment*. The first three entities store the fundamental information about assessors, topics and documents. The *task* keeps the information about a topic that is assigned to a user, the final ranking results and some information related to user interface features that we explain further in section 3.2.1. The *judgment* entity keeps a single pair of documents related to a specific task, the user’s action, and some important information about the algorithm state.

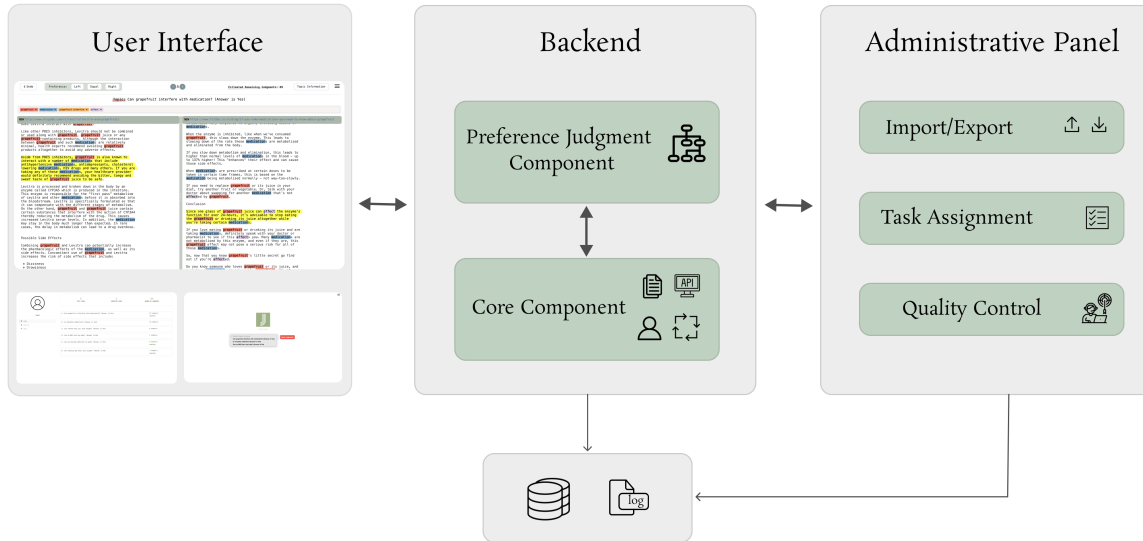


Figure 3.1: An overview of the JUDGO interface.

The user interface is comprised of three main pages: the home page, which is used for selecting assigned tasks; the profile page, which is used for presenting some static information about the user, tasks and judgments, and the judgment page, which is the main page in the JUDGO system and used for reviewing documents for each topic in pair form. The judgment page will be discussed in further detail in section 3.2.1.

The backend component is divided into preference judgment and core components. The first component keeps the suggested algorithm for preference judgment behind the JUDGO system. It has direct interaction with the core component, which allows it to display the next pair of documents that need to be reviewed. Additionally, it keeps a list of data structures defined as max-heaps in order to keep track of the preferences of the documents and generate a ranking order for them based on those preferences. It will be discussed further in section 3.3. The core component is responsible for handling frontend APIs, coordinating users' actions, interacting with the preference judgment component and database, as well as managing users and tasks.

The administrative panel has three main components: the task assignment platform, the quality control module, and the import and export of information such as users, tasks, topics, and documents. The key purpose of quality control is to discover whether or not

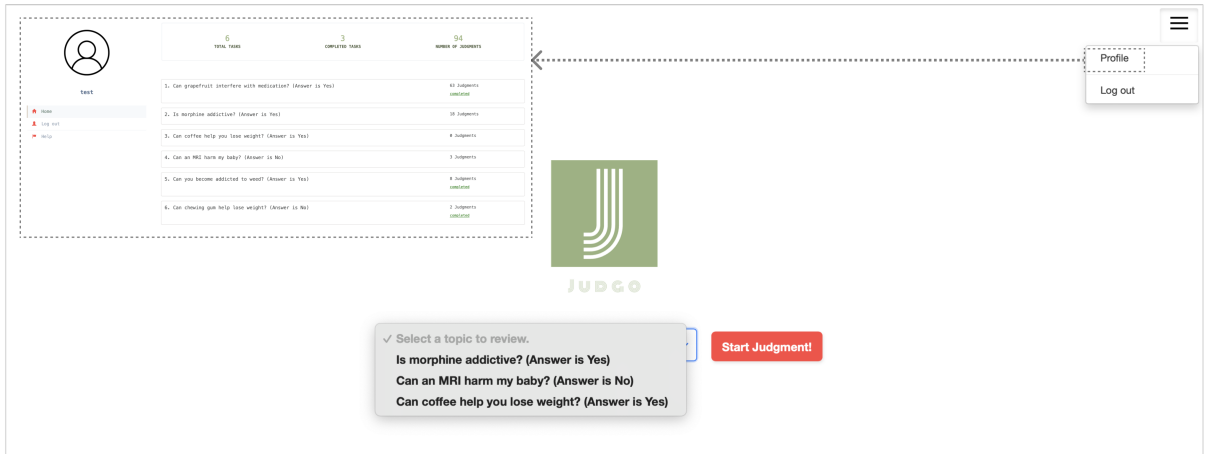


Figure 3.2: Detailed view of the home page.

the assessors maintain their level of consistency throughout the judgment process.

3.2.1 User Interface Side

In this subsection, we will review all the designed parts on the home page and then the judgment page, which is the main page of the JUDGO system, step by step from the assessor’s point of view. In addition, we describe in further detail each element of the system, including how it works, how each feature satisfies our initial design requirements and how it may be used.

Upon logging into the system, assessors are immediately redirected to the homepage. As shown in Figure 3.2, this page contains a list of topics that have been assigned to them, which they can select in any order based on their personal preference. In the top right-hand corner of the homepage there is a menu that has buttons for profile and logout; clicking on the profile button brings up the user’s profile page.

The profile page provides the evaluators with an overall summary of their activities, which enables them to determine how much of it they have done and how much of it they still need to do. It provides the total number of tasks, completed tasks, and total judgments that an assessor has performed up to this point, in addition to a list of tasks that specifies the title, current state, and the total number of judgments for each task.

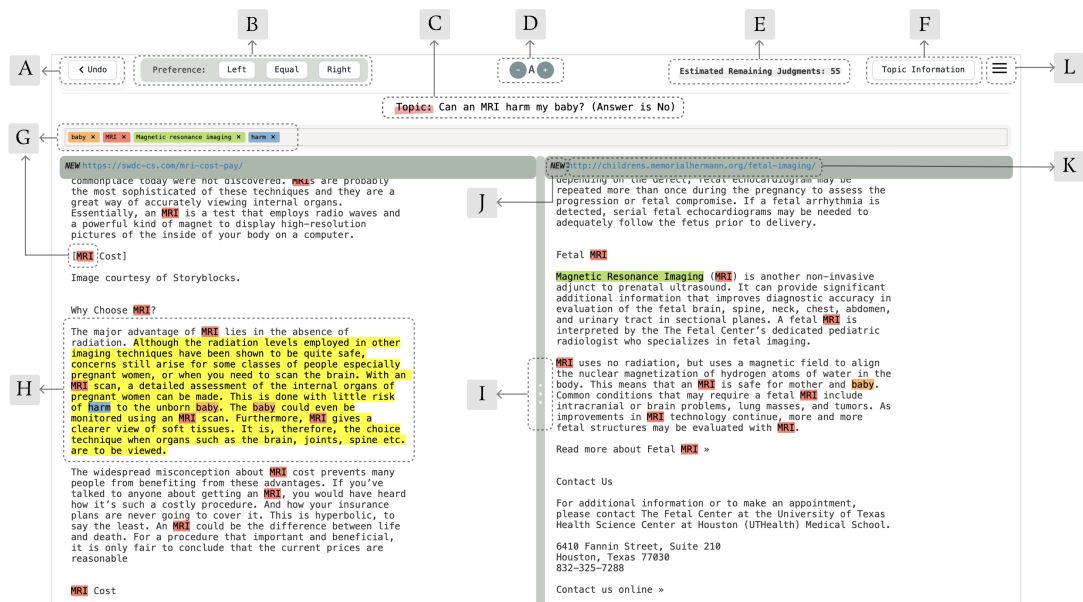



Figure 3.3: Detailed view of judgment page components.



After selecting a topic and clicking **Start Judgment!** button, the judgment page will be displayed. As illustrated in figure 3.3 and table 3.1, the judgment page is divided into a toolbar at the top of the page and document sections. The documents section comprises two documents, left and right, separated by a dragbar (I). Each document has an associated URL (K), title, and content.

At each step of the review process for a topic (C), evaluators have been provided with a unique pair of documents to assess, whereas the documents themselves may have been reviewed in the previous steps; in this scenario, the “new” label (J) will be displayed on the top of the document, right next to the URL. Assessors need to read both documents carefully, make a side-by-side comparison of the two, and then utilize the action panel (B) to determine which of the two documents is more relevant to the given topic (C). The “equal” button has to be used if the contents of the documents are identical or have the same level of preference.

As mentioned in design consideration (D2), assessors may lack sufficient expertise, making it difficult for them to determine how the presented documents are related to the given topic. In order to satisfy this requirement, the topic information button (F) is

provided; when this button  is clicked, a more detailed description of the topic is presented on the screen.

In order to meet the first design consideration (D1), which is related to increasing the speed of reading documents and making decisions, the JUDGO system is equipped with two essential features: the search box (G) and the highlight document (H). Both of these features give assessors the ability to sort out significant keywords and highlight specific sentences or paragraphs in the documents that are preserved throughout the evaluation process.

The (D4) system requirement is covered by the “Font Change” panel(D) , such that the assessors can modify the font size of documents based on their screen size, as well as the dragbar (I), which allows them to concentrate more intently on a particular document. In the subsequent subsection, we will elaborate on these three features. The undo button (A)  was designed in response to the initial demand (D3), which stated the ability for users to go back to reviews that they had previously finished and revise their decision.

Assessors are able to keep track of their progress as they move through the judgments by the estimated number of remaining judgments(E), which is displayed on the top right corner of the page next to the topic information button. In section 3.3.1, we explain how this number is calculated in more detail. Additionally, they can log out of the system at any time using the main menu (L) or return to the homepage to work on other topics that have been assigned to them. This can be done at any point in the judging process.

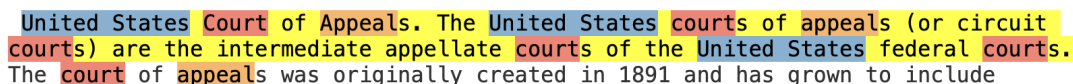
In the following, we will provide a more comprehensive explanation of four key features:

- **Search Box:** Assessors can use the search feature to enter keywords and find them in both documents. This component aids assessors in locating important keywords associated with the topics when they are unfamiliar with the subject or when the content of the documents is longer than expected. Since each keyword has a distinct colour associated with it, assessors can make conclusions quickly. Additionally, they are permitted to find up to 20 distinct terms or phrases with letters and numbers.
- **Mouse Highlighting:** A document may be presented to assessors multiple times, and each time, they will be asked to compare it to a different set of documents. This beneficial feature allows the user to highlight sentences and paragraphs in the document by clicking on them, dragging them around, and pushing the mouse up. They can delete the yellow-coloured highlight parts by choosing them once more. Any highlighted text will remain in the system until the end of the judging process. As

Table 3.1: List of features in judgment page.

	Feature Name	Description
A	Undo Button	It navigates to the previous judgment.
B	Action Panel	It has three buttons (left, right, equal) for the assessor to express their decision.
C	Topic Title	It displays the topic’s heading.
D	Change Font Panel	It enables assessors to adjust the font size of a document based on the device they are using.
E	Judgment Estimation	It is an estimation of the number of judgments remaining;
F	Topic Information	It provides a detailed description of the topic.
G	Search Box	It allows reviewers to search for up to 20 words and phrases in documents and highlight them in unique colours.
H	Mouse Highlighting	It allows reviewers to highlight relevant sections of the documents using a mouse.
I	Dragbar	It enables the assessor to focus on only one document by adjusting the width of documents on the left or right.
J	New Label	It indicates that the assessor has never seen this document before.
K	Document URL	It presents the URL of a website as the source for the document’s content.
L	Menu	It has options for logging out and returning to the home page.

shown in figure 3.4, when a user highlights a paragraph that contains several keywords coloured by the search feature, it handles the overlap between the highlighted parts by the mouse and the highlighted parts by the search keyword, which further helps the assessor make decisions.



United States Court of Appeals. The United States courts of appeals (or circuit courts) are the intermediate appellate courts of the United States federal courts. The court of appeals was originally created in 1891 and has grown to include

Figure 3.4: An example of overlap between the mouse-highlighted sentence in yellow and the search box-highlighted keywords

- **Change Font:** The task may be done by the assessors on various devices, including laptops, tablets, mobile devices, or large monitors. This feature enables them to change the size of the document to make the screen more personalized and easier to read documents. The newly adjusted size will remain in effect until the judging process is complete and will be reset when a new topic is selected for evaluation.
- **Drag bar:** On the judgment page, the drag bar is a user interface component located in the middle of the left and right documents. It gives users the ability to resize the width and re-organize left and right documents horizontally while simultaneously focusing on one side of the page. Users are able to obtain a more accurate picture of documents and URLs, which is helpful in situations with lengthy documents.

3.2.2 Administrative Panel Side

In this section, we provide a comprehensive explanation of the available features in the administration panel, focusing particularly on the component known as quality control.

The first component is the one that handles the uploading and importing of content. The admin panel provides a list of all tables currently designed in the database, together with the data they contain. We designed it in such a way that the administrator will have the opportunity to upload a list of users and tasks. This will allow assessors to have access to the data more rapidly than if it had to be entered manually one at a time.

Another component that the administrator may employ to assign a topic to a reviewer is called task assignment. However, they have the option of uploading a list of tasks via the importing tool as an alternative to manually assigning the tasks.

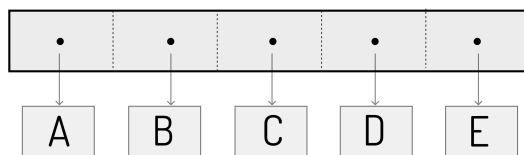


Figure 3.5: An example of the list of Pref object from a pool with five items: A, B, C, D, and E

One of the essential components designed for this system, which measures the accuracy of the assessors, is quality control [46, 39]. In some instances, reviewers are hired to review documents even when they lack expertise. As a result, we must assess their quality of judgment and gauge how consistent they are throughout the procedure. This feature also aids in determining whether or not assessors randomly hit the button.

This component selects a pair from previously finished judgments at random with a 10 percent chance after a certain number of judgments have been completed in order to measure quality. After that, the documents on the left and right switched places, and the new pair is presented to the reviewers.

According to their most recent action, if they choose the documents that they had not chosen before, it indicates that they are either not being consistent with their decision or are not paying sufficient attention to the task at hand. Throughout the process, the ratio of correct tests to the total number of tests will be displayed in the administration panel. Administrators can monitor this number and send a warning message via email to users whose totals fall below a predefined threshold.

3.3 Preference Judgment Algorithm

The JUDGO system uses a preference judgment algorithm that utilizes a tournament-style approach with several rounds. In each round, one or more documents are selected as winners and assigned to a specific ranking level. The selection process is based on user preferences and is facilitated by a heap-like data structure called Pref. The algorithm is based on previous research studies by Clarke et al[21].

The Pref data structure is similar to a max heap, it has an attribute called *topItem*

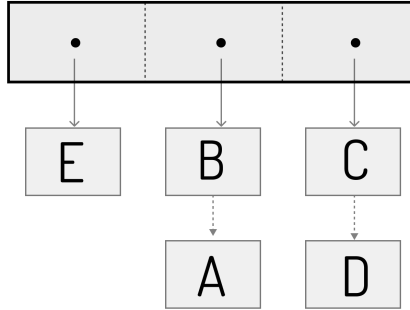


Figure 3.6: An example of the list of Pref objects with five items and two sequences of judgment, such as $(B > A)$ and $(C > D)$.

that keeps the most preferred item and a list of Pref objects, called *childrenList*, that hold all other items that are less preferable than the top item. The *topItem* attribute is used to compare the documents, and the *childrenList* is used to keep track of the less preferred items in each round of the tournament-style selection process.

The preference judgment algorithm, as shown in Algorithm 1, takes a pool of documents and a threshold number as input. The threshold number indicates how many documents should be retrieved and when the algorithm should stop. The first line of the algorithm uses the *buildPrefList* function to construct a list of Pref objects from the pool. Each Pref object has a document as its top item and an empty children list. For example, if the pool of documents includes A, B, C, D, E , the output of the *buildPrefList* function, as shown in Figure 3.5, would consist of a list of five Pref objects, one for each document in the pool. In the second line of the algorithm, a list of document pairs is also considered for keeping track of ties pairs.

The algorithm stops either when there are no items left in the list of Pref object or when the number of retrieved documents is greater than the pre-defined threshold, as can be seen in line 4 of Algorithm 1. In the next line, if there is only one Pref object left in the list of Pref objects, one round of the algorithm is completed, and a collection of documents as winners can be extracted and added to *rankedDocList*.

From line 6 to 10, the first and second Pref objects are removed from the *prefList* and their top item are presented to the user as left and right documents, respectively. The

Algorithm 1 The Preference Judgment Algorithm

Input

Pool: A list of documents.

K: A threshold for the number of top-retrieved documents.

Output

rankedDocList: A list of top retrieved documents.

```
1: prefList  $\leftarrow$  buildPrefList(Pool)
2: tieList  $\leftarrow$  empty
3: rankedDocList  $\leftarrow$  empty
4: while rankedDocList.size < K or not prefList.empty do
5:   while prefList.size > 1 do
6:     firstPrefObj  $\leftarrow$  prefList.pop(0)
7:     secondPrefObj  $\leftarrow$  prefList.pop(0)
8:     leftItem  $\leftarrow$  firstPrefObj.topItem
9:     rightItem  $\leftarrow$  secondPrefObj.topItem
10:    action  $\leftarrow$  getUserPreference(leftItem, rightItem)
11:    if action is right then
12:      secondPrefObj.children.append(firstPrefObj)
13:      newPrefObj  $\leftarrow$  secondPrefObj
14:    else if action is left then
15:      firstPrefObj.children.append(secondPrefObj)
16:      newPrefObj  $\leftarrow$  firstPrefObj
17:    else
18:      tieList.append((leftItem, rightItem))
19:      firstPrefObj.children.appendAll(secondPrefObj.children)
20:      newPrefObj  $\leftarrow$  firstPrefObj
21:    prefList.append(newPrefObj)
22:    rankedDocList.append(getBestAnswer(prefList, tieList))
23:    prefList  $\leftarrow$  prefList.pop().children
24: return rankedDocList
```

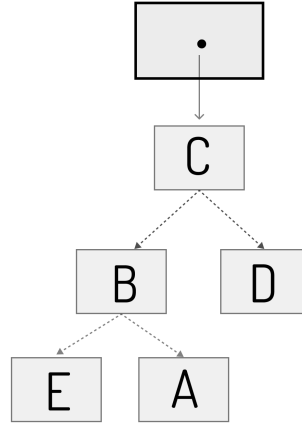


Figure 3.7: An example of the list of Pref objects. with five items and four sequences of judgment, such that $(B > A)$, $(C > D)$, $(B > E)$ and $(C > B)$.

getUserPreference function processes the user’s decision, which can be one of the three values: “left,” “right,” or “equal”. If the user prefers the right document, the *firstPrefObj* is then appended to the children list of the *secondPrefObj*, and the *secondPrefObj* is considered as a new Pref object. So in the *newPrefObj*, the right item has a higher priority over all its children. However, if the user prefers the left item over the right one, the *secondPrefObj* is added to the *firstPrefObj* children list (lines 11-16).

If the user decides that left and right items are ties, a new pair of the left and right items is added to the *tieList*. Following that, all children of *secondPrefObj* are added to the children of the *firstPrefObj*, and the right item is removed from consideration in further comparisons (lines 17-20). This means that the children of the *secondPrefObj* will be considered as less preferable than the left item (the *firstPrefObj*) in the next rounds of comparison. After processing the user’s decision, the *newPrefObj* is appended to the end of the *prefList* (line 21).

The figure 3.6 shows the process of creating a Pref object structure, which is used to represent the hierarchy of preferences between items. In the first step of the algorithm, the assessor compared items A and B and determined that B was superior to A. As a result, a new Pref object is created with B as the top item and A in its children list. In the second step, the user compared items C and D and preferred C over D. This resulted in the creation of another Pref object with C as the top item and D in its children list. The algorithm continues to build the Pref object structure as the user continues to compare

and evaluate items.

In the third step of the algorithm, the user is presented with E as the top item in the first Pref object and B as the top item in the second Pref object. If the user selects B over E and subsequently chooses C over B , the list of Pref objects will be updated to reflect these choices. The final structure will likely be as follows as it shown in Figure 3.7: The topmost element will be C , which is the final choice of user in the third step, B will be a child of C , as B was chosen over E but subsequently lost to C , E and A will be children of B .

In Figure 3.7, the first round of the algorithm is completed when there is only one Pref object in the *prefList*, line 5 in Algorithm 1. The function *getBestAnswer*, in line 22, is then used to extract the top documents in this round, by taking the top item in the single Pref object left in *prefList* and checking for any ties in *tieList*. The output of this function is then appended to the final *rankedDocList*. To proceed to the next round, the children of the single item in *prefList* are assigned to *prefList*, as shown in Figure 3.8, allowing the algorithm to continue evaluating the remaining documents.

In the first step of the second round, if the assessor decides that document B is tied with document D , a pair of (B, D) is added to the *tieList* and the second round of algorithm is finished since there is one Pref object in *prefList* again. Therefore, both documents are considered as the second-best set of documents in the pool. In the final round of the algorithm, there are documents A and E left to judge, and if A is chosen over E , no items remain in *prefList*, and the preference judgment algorithm for the given pool is completed. The final ranking of the documents would be the output of the algorithm, as determined by the assessor's preferences and any ties that were identified during the assessment as follows:

1. {Document C }
2. {Document B , Document D }
3. {Document A }
4. {Document E }

3.3.1 Number of Judgment Estimation

As mentioned in section 3.2.1, the assessors are provided with the estimated number of remaining judgments. To calculate this number, first, the total number of judgments should

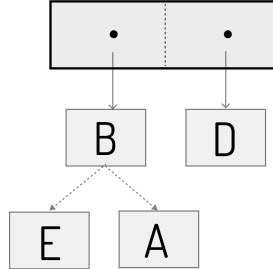


Figure 3.8: An example of a list of Pref objects when the first round of the algorithm is completed.

be calculated, considering there are N items to review in the given topic. In the first round of the judgment process, a user is required to judge $N - 1$ times to create the a single Pref object data structure and identify the items with the highest preference among the rest.

After the root has been extracted from the Pref object in the first round, it will be broken into several Pref objects, and the assessor should compare the top items of Pref objects in the next round. If there should be exactly top- k items to be extracted, so the user has to create one single Pref object from several Pref objects $k - 1$ times after the first round, and in the worst case, the number of Pref objects will be $\lceil \log(N - 1) \rceil$, and the user needs to compare them to find the next preferable documents. Therefore, the total number of judgments would be as follows:

$$Total\ Number\ of\ Judgment = (N - 1) + (k - 1) * \lceil \log(N - 1) \rceil$$

Chapter 4

Experiments on TREC 2022 Health Misinformation Track

The proposed preference judgment tool was employed in the TREC 2022 Health Misinformation track¹ to extract the optimal ranking for documents in each topic. In this chapter, we will explain TREC Health Misinformation in detail. Then, we elaborate on the topics and the documents as our dataset. Next, we discuss a detailed explanation of the deployment setting. We conclude with results, statistics regarding the users, the tasks, and the judgments and further feedback and discussion.

4.1 TREC 2022 Health Misinformation track

With the rapid growth of the internet worldwide, many people have used web searches to explore remedies or advice for various health-related topics. Many websites, however, contain misinformation, which means their content is not credible for the people’s initial need and might be written by people who believe in its correctness [58].

The primary goal of this track, which was formerly called Decision Track in 2019, is to encourage researchers to propose new retrieval methods which rank reliable and accurate information above misinformation[18]. TREC Health Misinformation Track in 2022 compromised two main tasks; core and auxiliary. The first is “web retrieval”, where participants must suggest a search method to prioritize credible and correct information over

¹<https://trec-health-misinfo.github.io>

incorrect ones to help people with their final decision. The second one is “Answer Prediction”, where participants need to predict the correct answer to the topic question, which can be yes or no.

The assessors, who are employed by the National Institute of Standards and Technology (NIST), evaluated the provided topics in three steps. The first step was to measure how much a document is useful for make a decision about topic’s question in order to eliminate the number of documents for the next step and there were three levels of usefulness: “Not Useful,” “Useful”, and “Very Useful”. In the second step, they had to decide what the document says is the right answer to the question for all “Useful” and “Very Useful” documents, and they had three options for answers including “yes”, “no”, or “unclear”. Finally, they were supposed to use our preference judgment tool to find the top-10 best documents that had the correct answer from “Very Useful” documents in the previous step.

4.2 Dataset

After completing the first and second phases of the judgment, topics, documents, and a mapping between them were available to ingest in the JUDGO tool.

- **Topics:** The track organizers had collected fifty topics, each containing a unique ID, query, question about a specific treatment or suggestion for a health issue, background and yes or no answers. You can see some examples in table 4.1. As we mentioned in the previous section, assessors judge topics in three steps. When all topics were judged in the two first steps, 38 topics were passed into the system. However, some of them had less than two relevant documents, which we ignored since there should be at least two “Very Useful” documents for preference judgment. We concatenated the question of the topic and its answer as the title of the topic in order to show them in the JUDGO system. Table 4.2 shows topics information such as title, number of documents and ID.
- **Documents:** English documents were extracted from the noclean version of the C4 dataset², which Google used to train the T5 model³. The collection is an April 2019 snapshot of Common Crawl⁴. There is more information about document collection on the Track website. Each document in the collection that is provided for the

²<https://huggingface.co/datasets/allenai/c4>

³<https://www.tensorflow.org/datasets/catalog/c4>

⁴<https://commoncrawl.org>

last phase of judgment contains UUID, title, content and URL. The URL can be a beneficial indicator of whether the documents have a valid answer to the topic question or not. Assessors found the mapping between topics and documents in the two first steps of the review process. As a result, 1477 documents labelled as “Very Useful” and had the correct answer for topics were forwarded to the JUDGO system. Table 4.3 shows a sample of a document that was ingested into the system.

Table 4.2: Information about topics that is provided to the JUDGO system.

ID	Documents	Title
151	91	Do tea bags help to clot blood in pulled teeth? (Answer is Yes)
157	128	Can cancer be inherited? (Answer is Yes)
158	31	Are vaccines linked to autism? (Answer is No)
159	11	Can baking soda help to cure cancer? (Answer is No)
160	133	Are squats bad for knees? (Answer is No)
161	11	Do ACE inhibitors typically cause erectile dysfunction? (Answer is No)
162	97	Is morphine addictive? (Answer is Yes)
163	14	Can fruit juice increase the risk of diabetes? (Answer is Yes)
164	25	Do magnetic wrist straps help with arthritis? (Answer is No)
165	3	Are sit ups bad for you? (Answer is Yes)
167	50	Are there health benefits to drinking your own urine? (Answer is No)
170	18	Can fish oil improve your cholesterol? (Answer is No)
171	6	Can you become addicted to weed? (Answer is Yes)
173	146	Is hydroquinone banned in Europe? (Answer is Yes)
174	19	Do men get UTI infections? (Answer is Yes)
175	20	Are carrots good for your eyes? (Answer is Yes)
176	7	Is methanol poisonous? (Answer is Yes)
177	21	Can an MRI harm my baby? (Answer is No)
178	152	Can exercise lower cholesterol? (Answer is Yes)
179	3	Can chewing gum help lose weight? (Answer is No)
180	12	Can sunglasses help prevent cataracts? (Answer is Yes)
181	85	Did AIDS come from chimps? (Answer is Yes)
182	14	Is too much water bad for you? (Answer is Yes)
183	2	Can HIV be transmitted through sweat? (Answer is No)

185	51	Can a woman get pregnant while breastfeeding? (Answer is Yes)
186	14	Can statins cause permanent cognitive impairment? (Answer is No)
187	45	Does Vitamin C prevent colds? (Answer is No)
188	7	Can coffee help you lose weight? (Answer is Yes)
189	2	Does drinking lemon water help with belly fat? (Answer is No)
190	38	Does deli meat increase your risk of colon cancer? (Answer is Yes)
191	23	Are skin tags contagious? (Answer is No)
192	19	Can oil pulling heal cavities? (Answer is No)
193	31	Does a high fiber diet help with hemorrhoids? (Answer is Yes)
194	44	Can grapefruit interfere with medication? (Answer is Yes)
195	25	Can vape pens be harmful? (Answer is Yes)
197	2	Is wifi harmful for health? (Answer is No)
199	34	Does ginger help with nausea? (Answer is Yes)
200	42	Can a cold sore cause genital herpes? (Answer is Yes)

4.3 Run-time Settings

When it comes to deploying and managing our tools, we took advantage of Heroku⁵, which is a container-based platform. It serves as a middleman between infrastructure and software and is a platform as a service (PaaS). All Heroku applications use Amazon Web Service (AWS), Infrastructure as a Service (IaaS), while Salesforce, a Software As Service (SaaS), supports them.

Based on the size of topics, documents and number of assessors, we selected a hobby dyno⁶ that is a container used at Heroku. In fact, dyno is a Linux container which provides memory, OS and filesystem virtually. There are various dyno types; the “hobby-1”, is suitable for small projects, contains 512MB RAM, and can be deployed from GitHub and supports ten different process types.

We used a Postgres “Standard 0”⁷ on Heroku as a database add-on. It has 4 GB of RAM, 64 GB of storage capacity, 25 backups, a limit of 120 connections, an unlimited number of rows, and an effective dashboard for monitoring. Since preference judgment is the last phase in the Track 2022 judgment process, during the experiment, we had to

⁵<https://www.heroku.com/home>

⁶<https://www.heroku.com/dynos>

⁷<https://elements.heroku.com/addons/heroku-postgresql>

Table 4.1: Sample of topics in TREC 2022 Health Misinformation Task.

Does deli meat increase your risk of colon cancer?
Background: Deli meats are meats that are processed to increase their storage life via curing or other methods and are commonly also known as lunch meats or cold cuts and used for sandwiches. Colon cancer is cancer of the colon (large intestine). The question is asking if the consumption of deli meat will increase the risk that a person develops colon cancer.
Answer: Yes
Do bananas increase the risk of diabetes?
Background: Bananas are a fruit. Diabetes is a disease that affects a person's ability to use sugar. The question is asking if consumption of bananas as part of one's diet will increase one's risk of developing diabetes.
Answer: No
Can coffee help you lose weight?
Background: Coffee is a commonly consumed drink made using hot water and ground coffee beans. This question is asking if consuming coffee, in some way, could aid weight loss.
Answer: Yes
Does drinking apple cider vinegar help lose weight?
Background: Apple cider vinegar (ACV) is a type of vinegar used in cooking. The apple cider vinegar diet involves consumption of a couple of tablespoons of ACV each day. This question is asking if the ACV diet helps people lose weight.
Answer: No
Can grapefruit interfere with medication?
Background: This question is asking if eating grapefruit and taking medication at the same time can have negative consequences because the grapefruit affects the function of the medication.
Answer: Yes
Is wifi harmful for health?
Background: WiFi is a family of wireless network protocols, which are commonly used for local area networking of devices and Internet access, allowing nearby digital devices to exchange data by radio waves. The question is asking if exposure to wifi radio waves could be harmful to health.
Answer: No

Table 4.3: Example of one document in TREC 2022 Health Misinformation Task.

UUID: en.noclean.c4-train.05939-of-07168.45060
Title: Chewing Mint Gum Doesn't Lead to Weight Loss
URL: https://www.redbookmag.com/body/healthy-eating/a15095/chewing-mint-gum-weight-loss/
Topic's Title: Can chewing gum help lose weight? (Answer is No)
<p>Content:</p> <p>Chewing Gum Might Make You Reach for the Chips</p> <p>A new study finds that chewing minty gum can actually lead to making poorer food choices.</p> <p>By Lauren Le Vine Mar 20, 2013</p> <p>Disappointing news for dieters who chew gum to aid weight loss and the entire REDBOOK staff: A new study in the journal Eating Behaviors researched the effects of chewing gum on food intake and found that it had no effect on the amount of calories consumed (and actually increased meal size in some cases), it also may lead to making poorer food choices. Chewing mint gum before meals made participants reach for foods like chips and candy instead of fruits or vegetables. Researchers attributed this to the effect menthol (the ingredient that makes gum minty fresh) has on taste buds — think about how an orange would taste right after brushing your teeth.</p> <p>There is a silver lining, though: This study only measured the effect of chewing gum before meals. Scientists wanted to debunk the pervasive and conflicting diet myths that chewing gum before a meal can either suppress your appetite or actually makes you hungrier because it gets digestive juices flowing. Chewing gum after a meal is still a satisfying way to cleanse your palate, and now we know why you won't want to eat again after your post-meal chew.</p>

wait for new topics and documents from the previous steps, which NIST should have done. In order to prevent data loss and conflict, we updated new entries during a defined time window (10 pm–12 am) and obtained a daily backup from the database.

Following the completion of the Heroku deployment, we added a total of eight users to the system. Two of these individuals were the organizers of the TREC 2022 Health Misinformation Track, and the NIST organizers were responsible for adding the remaining six users. In addition, we were provided with a username and password for each of the assessors. However, only six of them had fully completed the tasks. During the experiment, we stopped the judgment process as soon as the system had determined the rank of at least *ten* documents. Therefore, it was unnecessary for the reviewer to find the best possible ranking for all documents. However, for some topics, there are less than ten relevant documents; thus, assessors had to find the order of all documents.

4.4 Analysis

As discussed in section 4.3, we have six evaluators reviewing topics and documents that started working with the system between September 7 and October 26, 2022. During this window, one of the assessors that NIST had introduced decided not to participate before even completing a single topic judgment. Therefore, one of the track organizers completed the remaining tasks.

There were a total of 2266 judgments made by all assessors after they evaluated 38 different tasks. Assessors also had access to three powerful features within the JUDGO tool, including the ability to change the font size, search box, and mouse-highlighting documents. During TREC, all assessors altered the font size four times totally, entered 121 words and phrases in the search box and highlighted 90 documents by mouse.

The next parameter that we investigated was the amount of time that was allotted for the judging process. This is an important consideration for people who are interested in utilizing this tool because, in certain circumstances, evaluators are compensated, and it is, therefore, essential to accurately record the amount of time that has elapsed since the beginning of the assessment process.

Figure 4.1 illustrates the number of judgments that were completed within a time frame of less than 30 minutes. Any judgments that took longer than 30 minutes to complete were excluded from the data and considered outliers. As the figure clearly shows, the majority of the judgments were completed in less than 5 minutes.

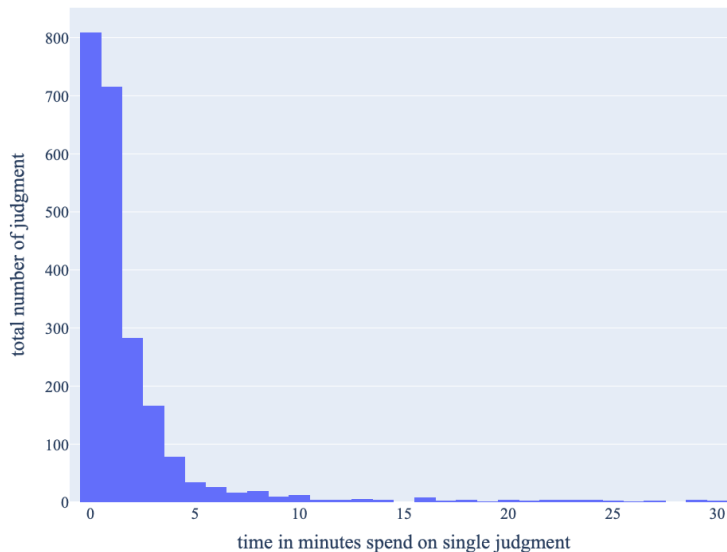


Figure 4.1: The total number of judgment completed in a time frame less than 30 minutes.

Besides the statistical information and the study of the evaluators’ actions, we will discuss the results of the evaluations for some topics. As we explained in section 3.3, each assessor’s choice significantly impacts the preference data structure. Thus, despite the fact that we stopped the judgment process as assessors found at least ten top documents, we cannot expect to have those ten documents ranked from one to ten. The reason for this is that, in some cases, several documents will stand in the same ranks since assessors evaluated them as being equal to one another.

For example, as seen in table 4.4, there are ten documents ranked from one to ten for topic ID 170, where document with rank 1 (UUID of “en.noclean.c4-train.02320-of-07168.77135”) is the preferable document that contains the exact answer about how fish oil cannot improve people’s cholesterol. It seems that throughout the judgment of these 10 documents, the assessors did not click the “equal” button, as there are no two documents that are in the same position in the ranking.

In contrast, in table 4.5, there are 11 documents that are ranked from one to four, with four documents placed in the first rank, two documents placed in the second rank, one document placed in the third rank, and four documents placed in the fourth rank. The fact that four different documents are placed in the first rank indicates that evaluators

Table 4.4: Result of document ranking for a topic ID 170 which is about Fish Oil.

Topic's title: Can fish oil improve your cholesterol? (Answer is No)		
Rank	UUID	URL
1	en.noclean.c4-train.02320-of-07168.77135	http://www.cholesterolcholesterol.com/fish-oil-cholesterol-cholesterol.html
2	en.noclean.c4-train.06050-of-07168.48635	http://www.dietandfitnesstoday.com/cholesterol-in-fish-oil.php
3	en.noclean.c4-train.06319-of-07168.16458	https://www.selfgrowth.com/articles/Mercola30.html
4	en.noclean.c4-train.06576-of-07168.35403	https://www.betternutrition.com/seven-ways/health-cholesterol-without-drugs-statins
5	en.noclean.c4-train.00519-of-07168.33397	https://health.clevelandclinic.org/fish-oil-supplements-vitamins-wont-lower-your-cholesterol/
6	en.noclean.c4-train.00351-of-07168.45102	https://www.curejoy.com/content/side-effects-of-fish-oil/
7	en.noclean.c4-train.02463-of-07168.98432	https://www.healthline.com/health/high-cholesterol/fish-oil-vs-statins
8	en.noclean.c4-train.01489-of-07168.98593	https://shop.advantagenutrition.com/about-krill-oil-c460.aspx
9	en.noclean.c4-train.05371-of-07168.17410	https://ourhealthhomelife.com/krill-oil-make-this-omega-3-supplement-your-health/
10	en.noclean.c4-train.03380-of-07168.1256	https://jarretmorrow.com/2010/10/17/fish-oil-supplementation-improve-body-composition/

had the same level of preference for each of them, given the fact that these documents are preferable to the others.

Unfortunately, there are not many highlighted sentences or paragraphs in the top documents related to topics 170 and 163. As is evident from table 4.6, topic id 165 with the title of “Are sit ups bad for you? (Answer is Yes)” has only three “very-useful” documents. We extracted the paragraphs that were highlighted by assessors during the judgment process. The document in the first rank specifically elaborates on how doing sit-ups affects your neck negatively. In contrast, the document in the second rank didn’t explain why sit-ups are bad for the body and explained just some facts and the document in the third rank focused on crunches, not sit-ups.

4.5 Feedback and Discussion

Following the completion of the TREC studies, we were provided with feedback regarding the functionality of preference algorithms. When an assessor is presented with a pair of documents ($d1, d2$), it takes considerably more time to view the document that they prefer again in the judgment process, especially if there are a large number of documents in the pool. As a result, the assessor may have to re-read the document again to recall its content and make a decision.

In addition, based on the analysis of features and the frequency with which users make use of those components, we address the possibility that some of these features will require revision in the future. As shown in table ??, The mouse-highlighting feature, which we initially claimed would save users’ time during the judgment process, has not been used frequently. As a result, we concluded that the functionality of this feature needed to be improved in future versions in order to make it more accessible to all different kinds of reviewers.

Table 4.5: Result of document ranking for a topic ID 163 about Fruit Juice.

Topic's title: Can fruit juice increase the risk of diabetes? (Answer is Yes)		
Rank	UUID	URL
1	en.noclean.c4-train.04920-of-07168.68393	https://tinyurl.com/bmjcontent
1	en.noclean.c4-train.00687-of-07168.89515	https://www.voice-online.co.uk/article/study-claims-fruit-juice-linked-type-2-diabetes?quicktabs_nodesblock=2
1	en.noclean.c4-train.00390-of-07168.99916	https://tinyurl.com/diabetesjournal
1	en.noclean.c4-train.04037-of-07168.63721	http://www.informationaboutdiabetes.com/articles/diet-and-nutrition/juiced-or-whole-discover-the-risks-of-drinking-fruit-juice
2	en.noclean.c4-train.03143-of-07168.87081	http://www.sick-celebrities.com/diabetes-2/whole-fruits-protect-against-diabetes-but-juice-is-risk-factor-say-researchers/
2	en.noclean.c4-train.05048-of-07168.997	https://www.nhs.uk/news/diabetes/fruit-juice-and-type-2-diabetes/
3	en.noclean.c4-train.06319-of-07168.141396	https://www.bmj.com/content/351/bmj.h3576.full
4	en.noclean.c4-train.06248-of-07168.132345	http://www.quantumday.com/2013/08/whole-fruit-diet-of-blueberries-grapes.html
4	en.noclean.c4-train.06198-of-07168.98632	http://guide2herbalremedies.com/regular-intake-orange-juice-is-linked-to-increased-diabetes-risk-women/
4	en.noclean.c4-train.00674-of-07168.115940	https://befitagain.com/does-type-2-diabetes-juicing-mix/
4	en.noclean.c4-train.03958-of-07168.63284	https://forums.sherdog.com/threads/fruit-juice-consumption-increases-risk-of-diabetes.774559/

Table 4.6: Result of document ranking for a topic ID 165 which is about SIT UP.

Topic's title: Are sit ups bad for you? (Answer is Yes)			
Rank	UUID	URL	Highlighted Paragraph
1	en.noclean.c4-train.04754-of-07168.131097	https://www.livestrong.com/article/521739-my-neck-hurts-from-situps/	It's no wonder many people report that their neck hurts after sit-ups — when you perform a sit-up, your spine undergoes compression, putting pressure on the discs between your vertebrae. Over many repetitions, this compression can result in swollen or herniated discs, which can lead to neck strain with sit-ups. Any pain experienced while exercising can be a warning sign that something is wrong, and the neck pain associated with the movement should not be ignored.
2	en.noclean.c4-train.06021-of-07168.68412	http://thesportseagle.co.za/doing-sit-ups/	Sit-ups has been one of the mainstream core exercises for many years however as medical science advances in the understanding of how the body works it is now clear that this core exercise has many negative effects on the body, especially the lower back and neck. Here are just a few concerns related to the exercise and why you need to cut sit-ups out of your gym or exercise program.
3	en.noclean.c4-train.06407-of-07168.111376	https://www.youarestrongbydesign.com/5-reasons-to-avoid-crunches-sit-ups/	A 1995 study found sit-ups placed over 3,000N of force on the lower spine, which could cause herniated discs. Imagine willingly applying that force on any other part of your body through exercise? You wouldn't. If you fall for the crunch myth, you will end up in pain, either in your back, neck or elsewhere.

Chapter 5

The System Adjustments

5.1 Algorithm Improvements

As explained in chapter 3, at each step of the judgment process, a pair of documents $(d1, d2)$ is shown to an assessor, and the assessor has three options, including left, right, and equal. In the algorithm, a new Pref object is created based on the assessor's preferences and then appended to the end of the list of Pref objects. In the revised algorithm, as shown in algorithm 2 (Line 21), the new Pref object is appended to the beginning of *prefList* to expedite the processing of judgments. As a result, the assessors will see the document they previously selected on the left of the new pair on the next judgment step; they will only need to read the right document and decide. Given that they have already read the left document, the new judgment will be simpler and faster for them to do.

As shown in figure 3.5 in chapter 3, if there are five documents A, B, C, D, E , the algorithm will present the pair (A, B) as the first pair for preference judgment to an assessor. The left-hand side of the figure 5.1 depicts how the list of Pref objects will look after the new adjustments if they conclude that document A is more relevant than document B . Therefore, the new Pref object is appended to the beginning of the list, as opposed to the right-hand side of the figure, the previous version of the algorithm, in which the new Pref object is appended to the end of the list.

Based on the improved version of the algorithm, the user will be presented with the document pairs of (A, C) in the following judgment; since they already have a fresh memory of document A , they only need to read document C to make a decision, which will likely be completed more quickly. The left-hand side of figure 5.2 shows how the list of Pref

Algorithm 2 The Preference Judgment Algorithm with Improvement

Input

Pool: A list of documents.

K: A threshold for the number of top-retrieved documents.

Output

rankedDocList: A list of top retrieved documents.

```
1: prefList  $\leftarrow$  buildPrefList(Pool)
2: tieList  $\leftarrow$  empty
3: rankedDocList  $\leftarrow$  empty
4: while rankedDocList.size < K or not prefList.empty do
5:   while prefList.size > 1 do
6:     firstPrefObj  $\leftarrow$  prefList.pop(0)
7:     secondPrefObj  $\leftarrow$  prefList.pop(0)
8:     leftItem  $\leftarrow$  firstPrefObj.topItem
9:     rightItem  $\leftarrow$  secondPrefObj.topItem
10:    action  $\leftarrow$  getUserPreference(leftItem, rightItem)
11:    if action is right then
12:      secondPrefObj.children.append(firstPrefObj)
13:      newPrefObj  $\leftarrow$  secondPrefObj
14:    else if action is left then
15:      firstPrefObj.children.append(secondPrefObj)
16:      newPrefObj  $\leftarrow$  firstPrefObj
17:    else
18:      tieList.append((leftItem, rightItem))
19:      firstPrefObj.children.appendAll(secondPrefObj.children)
20:      newPrefObj  $\leftarrow$  firstPrefObj
21:    prefList.insert(0, newPrefObj)
22:    rankedDocList.append(getBestAnswer(prefList, tieList))
23:    prefList  $\leftarrow$  prefList.pop().children
24: return rankedDocList
```

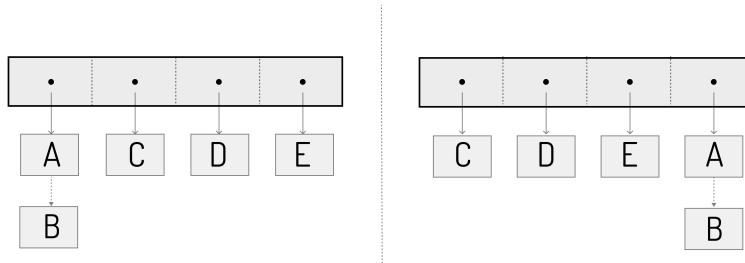


Figure 5.1: An example of two versions of the preference judgment algorithms after first judgment is done: the left-hand side shows the new version, and the right-hand side presents the previous one.

objects changes if users decide that document *A* is again more relevant than *C*. On the right side of the same figure, however, the user preferred *C* over *D*, two documents they had not previously viewed. So it is clear that the second version of the preference judgment algorithm provides assessors with a completely different experience. Except for the first judgement, assessors must read and interpret only one document from each judgement pair at a time in the second version because the user reviewed the other in the previous step.

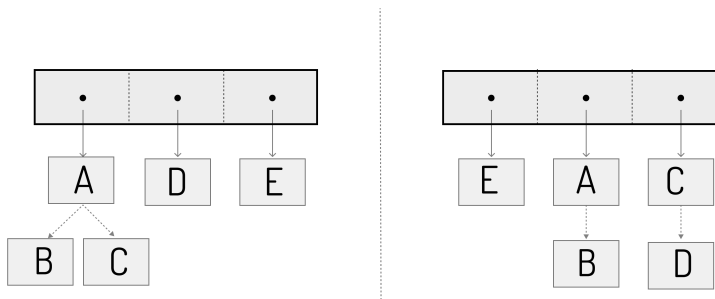


Figure 5.2: An example of two versions of the preference judgment algorithms after the second judgment is done: the left-hand side shows the new version, and the right-hand side presents the previous one.

Figure 5.3 demonstrates the two subsequent steps of the modified method; on the left, document *D* is chosen over document *A*. In the subsequent step, on the right, document *E* is selected over document *D* and one iteration of the algorithm is finished. As can be seen on the right, document *E* is the winner of the first round, document *D* is the second

best document, and document A is the third most preferable document. As a result, the assessors only need to evaluate one pair of B and C in the fourth iteration.

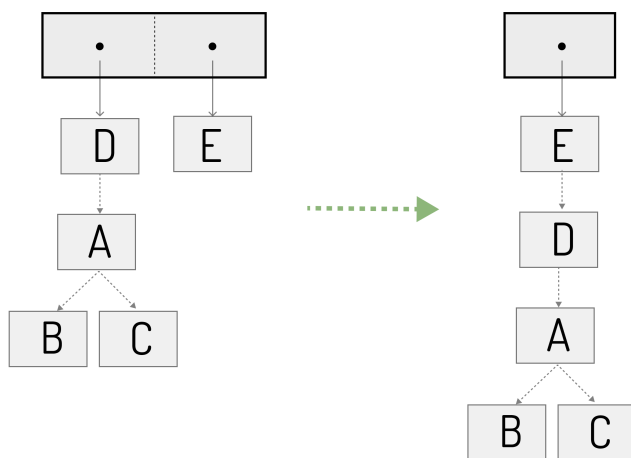


Figure 5.3: An example of two steps in the new version of the preference judgment algorithm, such that $(D > B)$ and $(E > D)$.

Chapter 6

Conclusion

In this thesis, we designed and developed a framework called JUDGO for preference judgments, which has been recognized as an alternative to absolute judgments. This system includes various components to satisfy the needs of both the researcher and the assessor and can be utilized to generate test data collection for offline evaluation of information retrieval systems. In addition, we suggested a novel preference judgment algorithm that focused on extracting the ranking order for the top-k documents. Our algorithm is structured similarly to a tournament, we allow ties and assume transitivity between pairs of judgments.

The JUDGO framework consists of three primary components, including the user interface, the backend, and the administrative panel. Within the user interface component, we carefully designed various features intending to accelerate the reading and decision-making processes. The administrative panel has two others to manage documents, queries, ranking results, and task assignments, in addition to one significant component for quality control of judgments. The suggested preference judgment algorithm is in the backend component, which is the core component of the system.

Using JUDGO, we conducted preference judgments to identify the top-10 best documents in ranking order for the 38 topics of the TREC 2022 Health Misinformation Track. The collected data serves as a benchmark to evaluate the effectiveness of the various system suggested by participants in TREC. According to our analysis of 2,200 different preference judgment pairs, most assessments are completed in less than 5 minutes. Furthermore, assessors found the search box feature, which enables them to search keywords in both documents, useful, but they didn't take advantage of highlighting individual documents by mouse.

There are several potential areas of future work for this thesis; during the TREC 2022 Health Misinformation Track, we did not track the quality of judgment. I hope to conduct more experiments on quality control components by employing non-expert assessors. We also compare the results of the suggested algorithm to the results of previous approaches.

The open-source implementation of the JUDGO framework and a sample dataset is available in the following repositories: <https://github.com/judgo-system>.

References

- [1] James Allan, Ben Carterette, and Joshua Lewis. When will information retrieval be “good enough”? In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 433–440, 2005.
- [2] Omar Alonso, Daniel E Rose, and Benjamin Stewart. Crowdsourcing for relevance evaluation. In *ACM SigIR forum*, volume 42, pages 9–15. ACM New York, NY, USA, 2008.
- [3] Negar Arabzadeh, Alexandra Vtyurina, Xinyi Yan, and Charles LA Clarke. Shallow pooling for sparse labels. *Information Retrieval Journal*, pages 1–21, 2022.
- [4] Peter Bailey, Nick Craswell, Ian Soboroff, Paul Thomas, Arjen P de Vries, and Emine Yilmaz. Relevance assessment: are judges exchangeable and does it matter. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 667–674, 2008.
- [5] Maryam Bashir, Jesse Anderton, Jie Wu, Peter B Golbus, Virgil Pavlu, and Javed A Aslam. A document rating system for preference judgements. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 909–912, 2013.
- [6] Mike Bendersky, Xuanhui Wang, Marc Najork, and Don Metzler. Learning with sparse and biased feedback for personal search. 2018.
- [7] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.
- [8] Róbert Busa-Fekete, Balazs Szorenyi, Weiwei Cheng, Paul Weng, and Eyke Hüllermeier. Top-k selection based on adaptive sampling of noisy preferences. In *International Conference on Machine Learning*, pages 1094–1102. PMLR, 2013.

- [9] Ben Carterette and Paul N Bennett. Evaluation measures for preference judgments. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 685–686, 2008.
- [10] Ben Carterette, Paul N Bennett, and Olivier Chapelle. A test collection of preference judgments. In *SIGIR 2008 Workshop: Beyond Binary Relevance: Preferences, Diversity and Set-Level Judgment*, *SIGIR*, volume 8, pages 3–5, 2008.
- [11] Ben Carterette, Paul N Bennett, David Maxwell Chickering, and Susan T Dumais. Here or there. In *European Conference on Information Retrieval*, pages 16–27. Springer, 2008.
- [12] Praveen Chandar and Ben Carterette. Using preference judgments for novel document retrieval. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 861–870, 2012.
- [13] Praveen Chandar and Ben Carterette. Preference based evaluation measures for novelty and diversity. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 413–422, 2013.
- [14] Olivier Chapelle and Yi Chang. Yahoo! learning to rank challenge overview. In *Proceedings of the learning to rank challenge*, pages 1–24. PMLR, 2011.
- [15] Olivier Chapelle, Donald Metzler, Ya Zhang, and Pierre Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 621–630, 2009.
- [16] Xi Chen, Paul N Bennett, Kevyn Collins-Thompson, and Eric Horvitz. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202, 2013.
- [17] Charles LA Clarke, Chengxi Luo, and Mark D Smucker. Evaluation measures based on preference graphs. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1534–1543, 2021.
- [18] Charles LA Clarke, Saira Rizvi, Mark D Smucker, Maria Maistro, and Guido Zuccon. Overview of the trec 2020 health misinformation track. In *TREC*, 2020.
- [19] Charles LA Clarke, Mark D Smucker, and Alexandra Vtyurina. Offline evaluation by maximum similarity to an ideal ranking. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 225–234, 2020.

- [20] Charles LA Clarke, Alexandra Vtyurina, and Mark D Smucker. Offline evaluation without gain. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, pages 185–192, 2020.
- [21] Charles LA Clarke, Alexandra Vtyurina, and Mark D Smucker. Assessing top-preferences. *ACM Transactions on Information Systems (TOIS)*, 39(3):1–21, 2021.
- [22] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. Overview of the trec 2021 deep learning track. In *30th Text REtrieval Conference. Gaithersburg, Maryland, 2021*.
- [23] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. Overview of the trec 2019 deep learning track. *arXiv preprint arXiv:2003.07820*, 2020.
- [24] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. Trec cast 2019: The conversational assistance track overview. *arXiv preprint arXiv:2003.13624*, 2020.
- [25] Zhicheng Dou, Ruihua Song, Xiaojie Yuan, and Ji-Rong Wen. Are click-through data adequate for learning web search rankings? In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 73–82, 2008.
- [26] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM conference on recommender systems*, pages 101–109, 2019.
- [27] Hans-Peter Frei and Peter Schäuble. Determining the effectiveness of retrieval algorithms. *Information Processing & Management*, 27(2-3):153–164, 1991.
- [28] Dorota Glowacka et al. Bandit algorithms in information retrieval. *Foundations and Trends® in Information Retrieval*, 13(4):299–424, 2019.
- [29] Ahmed Hassan Awadallah and Imed Zitouni. Machine-assisted search preference evaluation. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 51–60, 2014.
- [30] Kai Hui and Klaus Berberich. Low-cost preference judgment via ties. In *European Conference on Information Retrieval*, pages 626–632. Springer, 2017.
- [31] Kai Hui and Klaus Berberich. Transitivity, time consumption, and quality of preference judgments in crowdsourcing. In *European Conference on Information Retrieval*, pages 239–251. Springer, 2017.

- [32] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.
- [33] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *ACM SIGIR Forum*, volume 51, pages 243–250. ACM New York, NY, USA, 2017.
- [34] Thorsten Joachims et al. Evaluating retrieval performance using clickthrough data., 2003.
- [35] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Transactions on Information Systems (TOIS)*, 25(2):7–es, 2007.
- [36] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. Unbiased learning-to-rank with biased feedback. In *Proceedings of the tenth ACM international conference on web search and data mining*, pages 781–789, 2017.
- [37] Saikishore Kalloori, Francesco Ricci, and Rosella Gennari. Eliciting pairwise preferences in recommender systems. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 329–337, 2018.
- [38] Gabriella Kazai, Emine Yilmaz, Nick Craswell, and Seyed MM Tahaghoghi. User intent and assessor disagreement in web search evaluation. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 699–708, 2013.
- [39] Jinyoung Kim, Gabriella Kazai, and Imed Zitouni. Relevance dimensions in preference-based ir evaluation. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 913–916, 2013.
- [40] Caitlin Kuhlman, Diana Doherty, Malika Nurbekova, Goutham Deva, Zarni Phyoo, Paul-Henry Schoenhagen, MaryAnn VanValkenburg, Elke Rundensteiner, and Lane Harrison. Evaluating preference collection methods for interactive ranking analytics. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–11, 2019.
- [41] Matthew Lease and Emine Yilmaz. Crowdsourcing for information retrieval. In *ACM SIGIR Forum*, volume 45, pages 66–75. ACM New York, NY, USA, 2012.

- [42] Yan Li, Hao Wang, Ngai Meng Kou, Zhiguo Gong, et al. Crowdsourced top-k queries by pairwise preference judgments with confidence and budget control. *The VLDB Journal*, 30(2):189–213, 2021.
- [43] Jimmy Lin. The neural hype and comparisons against weak baselines. In *ACM SIGIR Forum*, volume 52, pages 40–51. ACM New York, NY, USA, 2019.
- [44] Eddy Maddalena, Stefano Mizzaro, Falk Scholer, and Andrew Turpin. On crowdsourcing relevance magnitudes for information retrieval evaluation. *ACM Transactions on Information Systems (TOIS)*, 35(3):1–32, 2017.
- [45] Alistair Moffat and Justin Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems (TOIS)*, 27(1):1–27, 2008.
- [46] Shuzi Niu, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. Top-k learning to rank: labeling, ranking and evaluation. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 751–760, 2012.
- [47] Kira Radinsky and Nir Ailon. Ranking from pairs and triplets: Information quality, evaluation methods and query complexity. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 105–114, 2011.
- [48] Kevin Roitero, Alessandro Checco, Stefano Mizzaro, and Gianluca Demartini. Preferences on a budget: Prioritizing document pairs when crowdsourcing relevance judgments. In *Proceedings of the ACM Web Conference 2022*, pages 319–327, 2022.
- [49] Mark E Rorvig. The simple scalability of documents. *Journal of the American Society for Information Science*, 41(8):590–598, 1990.
- [50] Tetsuya Sakai and Ruihua Song. Evaluating diversified search results using per-intent graded relevance. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1043–1052, 2011.
- [51] Tetsuya Sakai and Zhaohao Zeng. Which diversity evaluation measures are “good”? In *Proceedings of the 42nd international ACM SIGIR conference on Research and Development in information retrieval*, pages 595–604, 2019.
- [52] Tetsuya Sakai and Zhaohao Zeng. Good evaluation measures based on document preferences. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 359–368, 2020.

- [53] Mark Sanderson, Monica Lestari Paramita, Paul Clough, and Evangelos Kanoulas. Do user preferences and evaluation measures line up? In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 555–562, 2010.
- [54] Ruihua Song, Qingwei Guo, Ruochi Zhang, Guomao Xin, Ji-Rong Wen, Yong Yu, and Hsiao-Wuen Hon. Select-the-best-ones: A new way to judge relative relevance. *Information processing & management*, 47(1):37–52, 2011.
- [55] Paul Thomas and David Hawking. Evaluation by comparing result sets in context. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 94–101, 2006.
- [56] Ellen M Voorhees, Donna K Harman, et al. *TREC: Experiment and evaluation in information retrieval*, volume 63. MIT press, 2005.
- [57] William Webber, Alistair Moffat, and Justin Zobel. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38, 2010.
- [58] Liang Wu, Fred Morstatter, Kathleen M Carley, and Huan Liu. Misinformation in social media: definition, manipulation, and detection. *ACM SIGKDD Explorations Newsletter*, 21(2):80–90, 2019.
- [59] Xiaohui Xie, Jiaxin Mao, Yiqun Liu, Maarten de Rijke, Haitian Chen, Min Zhang, and Shaoping Ma. Preference-based evaluation metrics for web image search. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 369–378, 2020.
- [60] Xinyi Yan, Chengxi Luo, Charles LA Clarke, Nick Craswell, Ellen M Voorhees, and Pablo Castells. Human preferences as dueling bandits. *arXiv preprint arXiv:2204.10362*, 2022.
- [61] Ziying Yang, Alistair Moffat, and Andrew Turpin. Pairwise crowd judgments: Preference, absolute, and ratio. In *Proceedings of the 23rd Australasian Document Computing Symposium*, pages 1–8, 2018.
- [62] YY Yao. Measuring retrieval effectiveness based on user preference of documents. *Journal of the American Society for Information science*, 46(2):133–145, 1995.
- [63] Zhaohui Zheng, Keke Chen, Gordon Sun, and Hongyuan Zha. A regression framework for learning ranking functions using relative relevance judgments. In *Proceedings of*

the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pages 287–294, 2007.

- [64] Dongqing Zhu and Ben Carterette. An analysis of assessor behavior in crowdsourced preference judgments. In *SIGIR 2010 workshop on crowdsourcing for search evaluation*, pages 17–20, 2010.