Quantifying Uncertainty due to Ties in Rank Correlation Coefficients

An algorithmic approach to computing the bounds of uncertainty

Andreas Tsatsanis[1]
Supervisor: Julián Urbano[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Andreas Tsatsanis
Final project course: CSE3000 Research Project
Thesis committee: Julián Urbano, Lilika Markatou

Quantifying Uncertainty due to Ties in Rank Correlation Coefficients

Andreas Tsatsanis
Delft University of Technology

Abstract

Rank correlation coefficients are a common tool for describing similarity between ordered data. This study examines the use of the popular coefficient Kendall's τ, specifically in the case where the rankings contain *tied items* that *should not be tied*. Ties in this case represent *uncertainty* in the ranking, induced by the system that produced it, usually due to effects such as missing information or loss of precision (rounding). We propose two variants, τ^{min} and τ^{max}, which represent the lowest and highest possible correlation over all ways of arbitrating tied items. Our contribution is a novel quadratic-time algorithm for computing an arbitration of ties which yields the extremal correlation values τ^{min}, τ^{max}. We formally prove the correctness of the algorithm for the original Kendall's τ, and we suggest an adaptation for weighted variants of τ, such as τ_{AP} by Yilmaz et al. and τ_h by Vigna. Empirical evaluation on both synthetic ranking pairs and TREC ad-hoc system outputs demonstrates that ties often induce wide intervals $[\tau^{min}, \tau^{max}]$, indicating that no single value can fully encapsulate the uncertainty in correlation. These wide intervals also appear in rankings where current methods of computing τ correlation in presence of ties, namely τ^a and τ^b, have values large enough (≥ 0.9) for researchers to use as evidence of strong correlation. This indicates that currently used methods may yield false positive results. By reporting τ alongside its uncertainty bounds τ^{min} and τ^{max}, researchers are able to make more informed decisions, by demonstrating the reliability of correlation in presence of uncertainty-induced ties.

CCS Concepts

• **Information systems** → **Evaluation of retrieval results**;
• **Mathematics of computing** → *Exploratory data analysis*.

ACM Reference Format:
Andreas Tsatsanis. 2025. Quantifying Uncertainty due to Ties in Rank Correlation Coefficients. Bachelor Thesis. Delft University of Technology, Delft, The Netherlands.

1 Introduction

The comparison of ranking systems often entails comparing the rankings[1] they produce, and deriving different conclusions from there. Their "similarity", often quantified by their "correlation", is calculated through different *rank correlation coefficients*. Some examples include Spearman's ρ [13], Kendall's τ [7], τ_{AP} by Yilmaz et al. [17], Vigna's τ_h [15], and *Rank Biased Overlap* (*RBO*) by Webber et al. [16]. Each of these coefficients attempts to encapsulate the concept of how similar two rankings are, however, since the concept of "similarity" can be defined in different ways, these coefficients do not represent "similarity" the same way.

Suppose one is comparing two document retrieval systems, with the hypothesis that they generally rank the documents in a similar way. To test this hypothesis, one may assemble a set of documents, and ask the two systems to rank them. Computing the correlation of the two produced rankings can give an informative numeric value, which may either support or contradict the hypothesis. However, there is a source of error that has not been considered: one of the ranking systems, faced with ordering two elements, does not know which to put above the other, and instead chooses to label these items as *tied*. The presence of ties in the produced rankings reflects an uncertainty in the system. For a correlation coefficient this is a problem: how should ties be represented in the correlation value? Once again, there is no objective answer.

Rankings that inherently contain ties do exist in the real world (such as the aforementioned example of document retrieval rankings), and we often want to be able to measure their correlation. These coefficients mentioned above give us only a number, and this number does not show how much information was lost due to ties. One method of *quantifying* the uncertainty caused by ties is to find the *upper* and *lower bound* that the correlation *could* have had, if the system chose some ordering instead of ranking items as tied. This idea of using bounds as a measure of uncertainty was introduced alongside the definition of RBO by Webber et al. [16], as the upper and lower bounds $RBO^{\max}$, $RBO^{\min}$. Webber et al. initially addressed uncertainty due to unseen items[2], but a few years later Corsi and Urbano [3] adapted the concept to instead quantify uncertainty due to ties. They proposed two variants of RBO, namely RBO^{low} and RBO^{high}, that together can indicate the magnitude of uncertainty introduced by ties. Since their paper has shown the usefulness of these bounds for RBO, a natural question that arises is whether the notion of bounds is applicable for other measures of rank similarity.

Kendall's τ is a popular non-parametric coefficient of rank similarity, which defines similarity as correlation, measured through *concordance*[3]. This definition has worked well in practice, and many other coefficients have built upon it, extending the coefficient with properties such as top-weightedness[4] [12, 15] or average-precision [17]. This class of coefficients, known as τ correlation coefficients, due to their wide range of uses cases, is the focus of this study on uncertainty. Currently there are no methods for quantifying uncertainty due to ties in τ correlation, and thus this research proposes the τ^{min} and τ^{max} variants for τ, akin to RBO's RBO^{low} and RBO^{high}.

The difficulty in quantifying uncertainty through the suggested bounds is primarily a computational one: how do we find the minimum and maximum possible values of τ across all *permutations* of tied elements? The trivial algorithm of *try all possibilities* has a worst-case runtime of $O(n!^2)$, which makes even a small-scale experimental evaluation impossible, let alone practical real-world use of these bounds. If computed efficiently, these bounds can serve as a useful statistical tool, akin to RBO^{low} and RBO^{high}, but for the widely used τ correlation coefficients.

Corsi and Urbano [3] have proposed two greedy algorithms for computing bounds of RBO in quadratic time. Unfortunately, they cannot be directly applied to τ, since RBO represents similarity through *overlap*, while τ does so through concordance, and those definitions do not have an obvious two-way translation between

[1] The *order* the systems assign to a list of items.

[2] In the case where you only have access to a prefix of the ranking for evaluation. This type of uncertainty does not arise in τ.

[3] See Definition 1.

[4] Think of search engines where the top few results are much more significant than later pages of results. Two systems that agree on the first page but disagree on the second are more similar than two that disagree on the first but agree on the second.

Unpublished Bachelor's Thesis, Delft University of Technology
Source code is available at https://github.com/andtsa/bsc_thesis.

each other. Therefore, a new set of algorithms is needed for the concordance-based τ correlation coefficients.

Contribution

This research aims to show that the uncertainty due to ties carries a statistical relevance, and to propose a method for quantifying it. Concretely, these are:

- a new algorithm for finding a permutation of tied items that yields the highest and lowest possible τ correlation value,
- a formal proof of correctness,
- a practical analysis of τ correlations compared to their uncertainty bounds, evaluated on both synthetic data and real (TREC) data[5].

2 Background and previous work

The question of "are these two random variables correlated?" is a truly fundamental problem in statistics. A brief review of the relevant literature surfaces many ways, or *coefficients*, for assigning this relationship a number. Auguste Bravais [2] first introduced in 1846 the concept that would later become the *correlation coefficient*, and since then every new proposed coefficient attempts to amend the limitations encountered with the previous one.

Thus the concept of ties enters the discussion first and foremost as a practical obstacle: ties naturally appear in many real-world rankings, for which we likely still want to be able to measure correlation. How to do so is not obvious. Not all ties have the same reason for appearing in rankings, and different coefficients (or variants of the same coefficient) *interpret* ties in significantly different ways.

2.1 Interpreting ties

Depending on the nature of a ranking, ties can mean different things. If we are ranking pole vault athletes by the maximum height they jumped in the Olympics, then two (or more) athletes who jumped the same height on their best attempt *should* be tied in the ranking. Because the heights are discrete (integers, measured in centimetres) and there is no (remotely meaningful) uncertainty in measurement, in this case there is no uncertainty. On the other hand, a search engine assigning a relevance score to each document, may assign the same score to two (or more) documents. This does not mean the documents are the same, nor that the user is truly interested equally in both, but that the engine does not know the user's real preference, and this is often an *uncertainty* that ought to be taken into account. The first case is addressed by Gazeel [6], and we work with the second case.

2.2 Kendall's τ

In 1938, M.G. Kendall introduced the τ coefficient of rank correlation [7], and in 1945 two variants, τ^a and τ^b, to be used in presence of ties [8]. While its "superiority" over other coefficients is ambiguous, its usefulness is undeniable, and its ability to handle ties makes it an excellent candidate for ordinal data naturally containing ties.

Kendall computed correlation through *concordance*, which measures how much the two rankings agree upon the relative order of items. Formally, it is defined as follows:

Definition 1 (Concordance)**.** In two rankings

$$R = (\mathcal{S}, \prec_R) \text{ and } B = (\mathcal{S}, \prec_B)$$

a pair of distinct items $i, j \in \mathcal{S}$ is called "concordant" if the have the same relative ordering in both rankings, and "discordant" if they have an opposite ordering. Formally:

$$\text{concordant}_{R,B}(i,j) = \begin{cases} \texttt{true}, & i \prec_R j \iff i \prec_B j, \\ \texttt{false}, & i \prec_R j \iff j \prec_B i. \end{cases} \quad (1)$$

[5] Data from the Text REtrieval Conference.

In the context of computing τ, concordance is more usefully defined using the sign function sgn as follows:

$$c_{R,B}(i,j) = sgn(R_i - R_j) \cdot sgn(B_i - B_j), \quad (2)$$

where X_y is the numeric index of y in ranking X, and sgn is defined as:

$$sgn(x) = \begin{cases} -1, & x < 0, \\ 1, & x > 0. \end{cases} \quad (3)$$

Thus, for two rankings R and B, Kendall's τ correlation is defined as:

$$\tau = \frac{\sum_{\{i,j\}} c_{R,B}(i,j)}{n(n-1)/2}, \quad (4)$$

where:

- $\sum_{\{i,j\}}$ is a sum over all unordered pairs of different items i, j in the item set of R and B. For example, if $R = < a, b, c >$ and $B = < b, a, c >$, the summation would consider the pairs $\{a, b\}, \{a, c\}, \{b, c\}$.
- n is the length of the rankings[6]
- $n(n-1)/2$, also seen as $\binom{n}{2}$, is the total number of unordered pairs of items in each ranking.

If the two rankings completely agree on how the items should be ordered, then $\sum_{\{i,j\}} c_{R,B}(i,j) = \binom{n}{2}$ and $\tau = 1$, and if they completely disagree, $\sum_{\{i,j\}} c_{R,B}(i,j) = -\binom{n}{2}$ and $\tau = -1$.

2.3 Weighted τ

Multiple weighed variations of Kendall's τ have been introduced since then, each with a different weight function, and for a different use case. Some notable examples include the *Average Precision*-based τ_{AP} by Yilmaz et al. [17], and Vigna's hyperbolic-weighted τ_h [15]. When the rankings do not contain ties, they can be generalised by Shieh's τ_w [12]:

$$\tau_w = \frac{\sum_{\{i,j\}} c_{R,B}(i,j) w(R_i, R_j)}{n(n-1)/2} \quad (5)$$

where $w : \mathbb{N} \times \mathbb{N} \to \mathbb{R}$ is a symmetric ($w(x,y) = w(y,x)$) weight function over the indices of elements i, j in the "reference" ranking R. This leads to an asymmetric correlation coefficient, that can then be made symmetric by averaging the two τ_w values, as shown in equation 6:

$$\bar{\tau}_w(R,B) = \frac{\tau_w(R,B) + \tau_w(B,R)}{2} \quad (6)$$

The definition we use generalises over Shieh's τ_w, by using a weight function:

$$w : \mathcal{S} \times \mathcal{S} \to \mathbb{R},\ w(i,j) = f(R_i, R_j, B_i, B_j) \quad (7)$$

for any $f : \mathbb{N}^4 \to \mathbb{R}$. This definition is nearly identical to that used by Vigna [15], with the only adjustments intended to allow for weight functions that result in symmetric correlations[7]. For brievety, we refer to the weight function described in equation 7 as $w(i,j)$.

2.4 Kendall's τ with ties

For the initial definition of Kendall's τ, two items could *only* be concordant *or* discordant, meaning that two items may never occupy the same rank (be *tied*). Ties require special care when computing correlation, and this requires answering questions with no immediate answer, such as whether two rankings with *all items tied* should be correlated. Variants τ^a, τ^b defined by Kendall [9], or the lesser known τ^c defined by Stuart [14], are possible solutions for handling ties in the unweighted case, while Vigna has shown

[6] Kendall's τ is only defined for *conjoint* rankings, meaning they contain the same elements, in a (possibly different) order. Since the two rankings have the same elements, they also have the same length.

[7] The restrictions by Vigna allow for τ_w to be computed in $O(n \log n)$, which is not necessary here since the algorithm we propose runs in $O(n^2)$ anyway.

how to compute the weighted τ_w with ties [15]. Details on their computation can be found in appendix A.

Being relatively easy to compute and still providing somewhat meaningful correlation values even in the presence of ties, the variants of Kendall's τ have been widely used since. However, if these variants were sufficient for all the interpretations of ties described in Section 2.1, this paper would not exist. Consider two document rankings by two IR systems:

$$R = < a, [b, c, d], e > \tag{8}$$

$$B = < [d, a, c], b, e > \tag{9}$$

We can calculate that $\tau^a(R, B) = \frac{5}{10} = 0.5$, $\tau^b(R, B) = \frac{5}{7} \approx 0.714$, and $\tau^c(R, B) = \frac{3}{5} = 0.6$. Given that τ^b is most often the only implementation available in software packages, the conclusion here is likely to be that R and B are correlated. In Section 2.1 we argumented why there is a case to be made for ties representing uncertainty, and here we can see its effects:

$$R_1 = < a, b, d, c, e >,\ B_1 = < c, d, a, b, e >,\ \tau(R_1, B_1) = \mathbf{0}$$

$$R_2 = < a, c, d, b, e >,\ B_2 = < a, c, d, b, e >,\ \tau(R_2, B_2) = \mathbf{1}$$

Different permutations of the tied items can yield very different correlations. Clearly the answer is not as simple as "R and B are correlated", and some additional context is needed.

2.5 Uncertainty due to ties in Rank Biased Overlap

The interpretation of ties as a product of *uncertainty in the ranking itself* is not new. Corsi and Urbano [3] not only discussed this concept in detail, but also proposed two algorithms, namely RBO^{low} and RBO^{high} for computing the optimal arbitration of ties such that overlap is minimised or maximised respectively.

Both of these algorithms work by linearly constructing the desired rankings, instead of computing the RBO value for different permutations. Their algorithms work by looking at a prefix of the rankings, checking if there are any ties at that depth, "fixing" items in place (i.e. removing them from their tie-groups and assigning a final position) such that overlap is minimised for RBO^{low} and maximised for RBO^{high}, and then extending the prefix. The reason this works is that in RBO any permutation of ties which maximises the overlap (between the two rankings) *as early as possible, also* maximises the final RBO value. This property did not emerge accidentally though. In the words of Webber [16]:

> [*RBO*] achieves this by using a convergent set of weights across successive prefixes, preventing the weight of the unseen tail from dominating that of the observed head.

While these algorithms may not be directly applicable, or adaptable, for computing the min/max bounds for τ, they could hint that a greedy approach may work for τ as well.

2.6 Orders and rankings

So far we have referred to rankings as *lists of items*, with some items being ranked higher or lower than others. For the rest of this paper, we use the formalisations described in this section to prevent any ambiguity.

In the definition of concordance (Definition 1) we already saw the symbol ($\prec_x$) for an *order*. We define what a ranking is based on the definition of an order:

Definition 2 (Order)**.** An order over some set $\mathcal{S}$ is a transitive binary relation[8] over $\mathcal{S} \times \mathcal{S}$. A *total* order is one where the binary relation $\prec$ is defined between every pair of distinct elements: $\forall a, b \in \mathcal{S} : a \neq b \iff (a \prec b \vee b \prec a)$. A binary relation is transitive if $\forall a, b, c \in \mathcal{S} : (a \prec b) \wedge (b \prec c) \implies (a \prec c)$.

[8] See https://en.wikipedia.org/wiki/Homogeneous_relation

In this paper, we always refer to one of two special cases of orders, namely *strict total orders* and *partial total orders*.

- **Strict total order**: a total order $\prec$ that is *irreflexive* ($\neg(a \prec a)$, i.e. no element is related to itself) and *asymmetric* ($a \prec b \iff \neg(b \prec a)$)
- **Partial total order**: a partial total order $\preceq$ is some total order that is not necessarily strict: i.e. it is possible that $\exists a, b \in \mathcal{S}$ such that $a \preceq b \wedge b \preceq a$.

Definition 3 (Ranking)**.** A ranking is an *ordered set*, or formally, a 2-tuple $R = (\mathcal{S}, \prec_R)$ of a set of items $\mathcal{S}$ and a total order $\prec_R$ over the elements in $\mathcal{S}$. Additionally:

- A ranking can contain ties, in which case its order is a partial total order, and denoted by $\preceq_R$.
- Two rankings that have the same items are called *conjoint*. Since the τ coefficients only operate on conjoint rankings, we only ever consider this case, and refer to the (shared) collection of items as the *item set* $\mathcal{S}$ of the rankings.

3 Methods

There are numerous ways to approach the problem of designing an algorithm for computing the bounds of τ uncertainty. First we discuss some notable ones, along with their limitations. Then in Section 3.2 we describe the chosen problem representation, followed by the tractability assumption in section 3.3.

3.1 Approaches

The space of all possible combinations of permutations is bounded by $O(n!^2)$, and thus a brute-force search is virtually impossible. While a thought could be to try to narrow the search space, possibly taking inspiration from pruning in SAT solvers, no search algorithm that finds *all* the optimal solutions can be used, because the *solution* space is also of superpolynomial size[9]. This eliminates a large class of search algorithms, and shows that any algorithm that relies on finding *all* solutions will not run in polynomial time.

Instead of searching for the permutations that yield maximum correlation, we can try *constructing* them such that they result in optimal correlation. This is the approach used for computing RBO^{low} and RBO^{high} [3]. While those algorithms have not been proven optimal, experimental results show that they likely are. Their approach examines the items in the rankings depth-by-depth[10] and greedily tries to un-tie them such that *overlap* is maximised. While this step by step (or depth-by-depth) construction could work for τ as well, there is one significant issue that prevents us from pursuing this approach directly: *concordance does not depend on depth*. As such, there is no clear start for deriving an algorithm to maximise concordance while examining the rankings depth-by-depth.

The final approach we considered is to represent the rankings as graphs. This approach is favourable because concordance is elegantly represented as a *set intersection of the two graphs' edge sets*, and the transitive property of rankings' orders translates to acyclicity in graphs. An initial approach was to reduce the problem to one solvable in exponential time by the algorithms shown and formally proven by Bodlaender et al. [1]. While a reduction from $O(n!^2)$ to $O(2^n)$ is a significant improvement, it is still prohibitively expensive for real-world use on data such as the TREC rankings, where a typical TREC run contains upwards of 1000 elements. Fortunately, taking inspiration from the greedy algorithms for RBO, we will now show how the graph representation allows us to derive an algorithm for maximising concordance in just $O(n^2)$.

[9] For the simplest case, consider two rankings with all elements tied. Pick any of $n!$ permutations for one ranking, and copy the order for the second ranking: those are $n!$ valid solutions for τ^{max}.

[10] Depth in RBO is the size of the prefix (of the full ranking) that we are currently looking at. At depth 1, we only see the first element in each of the two rankings, at depth 2 the first two elements, and so on.

3.2 Representing ranks as graphs

Abstractly, a ranking is an ordered set of items (Definition 3)

$$R = (\{a, b, c, d, e\}, \preceq_R) \tag{10}$$

A familiar representation of an ordered set is a *list* (in ascending order):

$$R = < a, b, c, d, e > \tag{11}$$

But we can equivalently represent it as a graph (Figure 1)

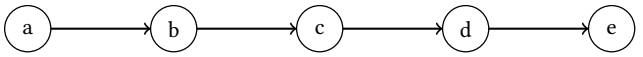


Figure 1: A graph representation of the ordered set R

In the above graph, a directed edge $\overrightarrow{xy}$ means that x is *immediately* before y. A more useful representation would be to have edge $\overrightarrow{xy}$ mean that $x \prec_R y$, or that in a *total ordering* $\prec_R$, x preceeds y. In that case, we draw one edge for every relation $x \prec_R y$ as seen in figure 2. Such a graph is a referred to as a *tournament*.

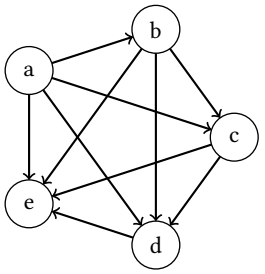


Figure 2: Graph representation of a total order

Definition 4 (Tournament)**.** A tournament graph is a directed graph $T = (V, E)$ with *exactly one edge connecting every pair of vertices*. Furthermore, an *acyclic tournament* is a tournament with no *directed cycles* in E, and a *transitive tournament* is a tournament where:

$$\forall a, b, c \in V : (\overrightarrow{ab} \in E \wedge \overrightarrow{bc} \in E) \implies \overrightarrow{ac} \in E.$$

The similarity between the ranking R and the tournament graph we constructed might lead us to suspect some stronger relation between the two concepts. We can actually show that this construction can be formalised through theorem 1.

Theorem 1. Transitive tournaments are equivalent to strict total orders. See Appendix B for proof[11].

Another observation is that there *are no cycles* in the constructed graph. This is not a special propety of this example, but a property of all transitive tournaments:

Theorem 2. Every acyclic tournament is a transitive tournament, and every transitive tournament is acyclic. See Appendix B for proof.

If we were to allow two edges between pairs of elements (breaking the definition of a tournament), we can allow the graph to represent *ties*: since two elements x and y being tied is equivalent to $(x \preceq_R y) \wedge (y \preceq_R x)$, we can represent it using a *cycle*. Suppose our previous ranking R had some tied elements:

$$R = < a, [b, c, d], e > \tag{12}$$

in this example b, c and d are all tied with each other. The left graph of Figure 3 shows the graph representation of R. We can see that every tie in the ranking is represented by a unique cycle in the graph.

[11]Theorems 1 and 2 can be considered general knowledge, and are by no means a novel contribution of this paper. Nevertheless, the proofs are ours, and we believe can provide some useful intuition for the curious reader.

The goal of finding a way to break the ties is now represented as breaking all cycles in the graph. But any arbitrary way to break cycles will not do: for each of the two rankings, we need to pick which edges to remove in such a way as to maximise (or minimise) the τ correlation of the resulting rankings without ties.

3.3 Maximising concordance

At this point, assumptions are required to generalise to variants of τ. Without additional constraints, maximising (or minimising) $\sum_{\{i,j\}} c(i,j)w(i,j)$ is (a variation of) the Knapsack problem[12], which is known to not be solvable in polynomial time. Similarly to how *RBO* maximises the final value by maximising overlap, we can maximise *correlation* by maximising *concordance*; which is the obvious approach for the *unweighted* variant τ, or τ_w with $w(i,j) = 1$. The weighted variant is discussed more in section 4.5.

As seen in Kendall's *Rank Correlation Methods* [8], the formula of τ (equation 4) can be rearranged as the following equation:

$$\tau = \frac{2C}{n(n-1)/2} - 1. \tag{13}$$

The objective of minimising or maximising τ correlation is to minimise or maximise concordance C (the number of concordant pairs). And here, C is equal to the number of edges in the intersection of the two graphs. To make this point more clear, suppose we have a second ranking with ties B. The graphs of our two rankings are shown in Figure 3.

$$R = < a, [b, c, d], e >$$

$$B = < c, [d, a], e, b >$$

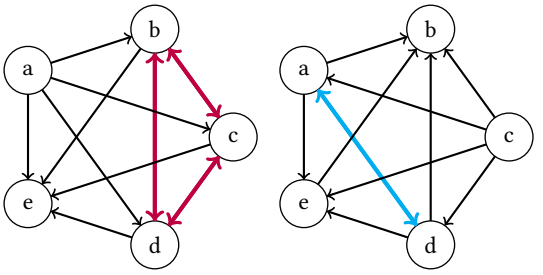


Figure 3: Graph of rank R (left), graph of rank B (right).

If we look at the elements a and b for example, we see that in both lists R and B, a appears before b: our definition of concordance. In graph terms, a being before b means there must be an edge from vertex a to vertex b. If both graphs have this edge, then the elements are concordant. In Figure 4 we have highlighted in green and orange all the edges corresponding to concordant and discordant pairs respectively.

Note that we never defined concordance between elements that are in ties. This may seem like a problem, since we do have ties, but in reality it does not matter, because what we really care about is *choosing the orders* that will maximise or minimise concordance.

Let us further break down this problem: a pair x, y can be neither concordant nor discordant in two cases:

- one of the graphs has both $\overrightarrow{xy}$ and $\overrightarrow{yx}$, while the other has only $\overrightarrow{xy}$ (or $\overrightarrow{yx}$),
- or both graphs have both $\overrightarrow{xy}$ and $\overrightarrow{yx}$.

In the first case, what if we *look into the other graph* to see whether we should pick $\overrightarrow{xy}$ or $\overrightarrow{yx}$? And in the second case, is the decision not arbitrary?

[12]https://en.wikipedia.org/wiki/Knapsack_problem

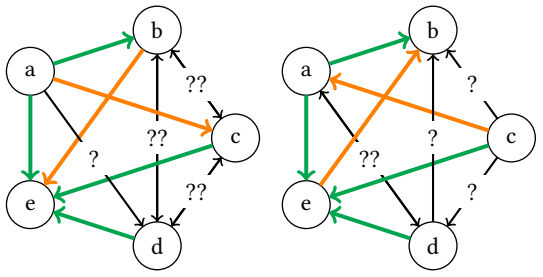


Figure 4: Graph of rank R (left), graph of rank B (right), each showing concordant pairs in green and discordant in orange. Notice that there are pairs neither concordant nor discordant. Sometimes a choice between two edges is required in one ranking, and sometimes in both. So the question is: which order do we pick?

4 Construction from graph representation

From two rankings R and B containing ties, we want to break the ties in such a way that we end up with two rankings without ties R', B', such that there is no other way to break the ties R'', B'' where $\tau(R'', B'') < \tau(R', B')$.

4.1 Starting point

Take the order representation of the two rankings,

$$R = (\mathcal{S}, \preceq_R), \quad B = (\mathcal{S}, \preceq_B) \tag{14}$$

We first construct the graph representation of the partial orders G_R, G_B, as described in Section 3.2. Algorithm 1 shows a pseudocode for this procedure.

Algorithm 1 Constructing a graph of a partial total order

```
1: procedure PartialOrderGraph(ranking X)
2:     G ← (ItemSet(X), ∅)
3:     for i = 0 to Count_tie-groups(X) do
4:         tg_1 ← i-th tie-group of X
5:         for j = i to Count_tie-groups(X) do
6:             tg_2 ← j-th tie-group of X
7:             for (x, y) ∈ tg_1 × tg_2 where x ≠ y do
8:                 G ← (V, E ∪ {x⃗y})
```

4.2 What does the solution look like?

We need to remove enough edges from both graphs G_R, G_B to break all cycles, but not so many that we lose the information about which element is supposed to go where in R', B'. The term for such sets of edges is known as a feedback arc set:

Definition 5 (Feedback Arc Set)**.** A feedback arc set of a graph $G = (V, E)$ is a set of edges $S \subseteq E$ for which the graph $G' = (V, E \setminus S)$ is *acyclic*.

A *minimum feedback arc set* (MFAS) of graph $G = (V, E)$ is a feedback arc set $F \subset E$ such that for *any* feedback arc set S of graph G, $|F| \leq |S|$ (F contains the fewest possible edges that break all cycles).

There are two observations we can make here, that together let us prove two more key theorems:

(1) We can make an observation about *where* these ties will be: as we showed before, every tie is a cycle and every cycle is a tie, so cycles are entirely contained within their tie-groups. This means tie-groups are *fully connected components* of the larger graph, and any edge in the feedback arc set will only ever impact a single component.

(2) The edges we need to discard are a feedback arc set. We can show that if and only if the resulting graph is a tournament, then we picked the smallest possible number of edges to remove (and our FAS is a MFAS).

Now we will show that for each connected component, we need to discard exactly half of the edges.

Theorem 3. Every minimum feedback arc set F of a complete directed graph $G = (V, E)$ with $n = |V|$ vertices will contain exactly $\binom{n}{2} = \frac{n(n-1)}{2}$ edges.

Proof. In a fully connected graph G there are $\sum_{k=2}^{n}(k-1)!\binom{n}{k}$ cycles, but only $2\binom{n}{2}$ edges, so clearly (for $n > 2$) we do not need to spend an edge for every cycle–*except for 2-cycles*. No edge that is part of a 2-cycle will be part of any other 2-cycle. Since a minimum feedback arc set must break every cycle, we know that it will need to contain at least one of the two edges of every 2-cycle, and hence we have a lower bound:

$$|F| \geq \binom{n}{2} \tag{15}$$

We will now find an upper bound to $|F|$. Pick any strict total order r of the vertices V: if we call E_r the set of edges that correspond to the order r, then:

$$\forall x, y \in V \,:\, x \prec_r y \iff \overrightarrow{xy} \in E_r \tag{16}$$

But what edges are in E_r? Since a strict total order does not support reflexivity, E_r will always contain exactly one edge for every pair of elements, or $\binom{n}{2}$ edges. This is exactly half of the edges in a complete directed graph, so we are left with a set of edges $E \setminus E_r = E_{\bar{r}}$. This set, we will call $E_{\bar{r}}$, has the interesting property that it is also a strict total order (which we will aptly refer to as $\bar{r}$):

$$\forall x, y \in V \,:\, \overrightarrow{xy} \in E_{\bar{r}} \iff \overrightarrow{yx} \in E_r \tag{17}$$

$$\equiv x \prec_{\bar{r}} y \iff y \prec_r x \tag{18}$$

What this means is that we can use either E_r of $E_{\bar{r}}$ as a feedback arc set, because we know from Theorems 1 and 2 that a strict total order is equivalent to an acyclic tournament. We have shown that $|E_r| = |E_{\bar{r}}| = \binom{n}{2}$, so any ***minimum*** *feedback arc set* can contain at most $\binom{n}{2}$ edges:

$$|F| \leq \binom{n}{2} \tag{19}$$

From Equation 15 and Equation 19 we get $|F| = \binom{n}{2}$. ■

Theorem 4. For every minimum feedback arc set F of a complete directed graph (CDG) $G = (V, E)$, the graph $G = (V, E \setminus F)$ represents a strict total order r (meaning that for every edge $\overrightarrow{xy}$ between $x, y \in V$, $\overrightarrow{xy} \in (E \setminus F) \iff x <_r y$)

Proof. We saw in the proof of Theorem 3 that every feedback arc set will need to remove one edge from every 2-cycle, leaving exactly one edge between every pair of nodes. This makes the resulting graph $T = (V, E \setminus F)$ a *tournament*. More specifically, an *acyclic tournament*, since the feedback arc set must break *all* cycles, meaning the edges removed from the 2-cycles must also break all 3-cycles ($A \to B \to C \to A$), 4-cycles, etc. From Theorem 2 we know that T is transitive, and with Theorem 1 we show that T represents a strict total order.
Therefore, for any MFAS F, a CDG with all the edges in F removed will represent a strict total order. ■

Because the final graph must be a tournament, the set of edges we need to remove are a minimum feedback arc set.

4.3 Construct final solution

It is both conceptually and computationally easier to think about whether *adding* an edge to an already acyclic graph will create a cycle, than whether removing an edge will eventually make it acyclic. Thus, we begin by constructing two empty graphs $G'_R = G'_B = (\mathcal{S}, \emptyset)$. Instead of directly finding feedback arc sets, we will solve the equivalent problem of picking which edges to keep. But which edges should we pick, in order to ensure maximum concordance? In the following explanation, the maximisation case is shown, and in Section 4.6 we explain how to invert the algorithm.

Theorem 5. *Picking an edge that increases concordance without creating a cycle is always optimal.* More precisely, when breaking some tie $[x, y]$ in one ranking, if the other ranking graph contains edge $\overrightarrow{xy}$ then picking the order $x \prec_R y$ for the first ranking will always yield at least as good a solution as picking $y \prec_R x$ (as long as the solution containing $\overrightarrow{xy}$ is valid, i.e. an acyclic tournament).

We try to maximise $|E'_R \cap E'_B|$ by adding as many edges from the intersection $|E_R \cap E_B|$:

- for every $e \in E_B$
  - if we *can* add this edge to G'_R, meaning $e \in E_R$, and that adding this edge does not violate the acyclic property of G'_R, then we add it.
- repeat for the other graph, adding edges *from* E_R *to* G'_B

4.4 Final step

As we proved in Theorem 2, the two acyclic tournaments G'_R, G'_B are transitive, and by Theorem 1 we now have two strict total orders, and that is our answer. In order to implement Theorem 1, any sorting algorithm with a lambda for comparing based on the graphs can be used, shown in the full pseudocode in Algorithm 2.

Algorithm 2 (Full) Graph algorithm for τ_{max} of rankings with ties R, B with item set $\mathcal{S}$

$G_R : (\mathcal{S}, E_R) \leftarrow$ PartialOrderGraph(R)
$G'_R : (\mathcal{S}, E'_R) \leftarrow (\mathcal{S}, \emptyset)$
$G_B : (\mathcal{S}, E_B) \leftarrow$ PartialOrderGraph(B)
$G'_B : (\mathcal{S}, E'_B) \leftarrow (\mathcal{S}, \emptyset)$
for edge $\overrightarrow{xy} \in$ `Sorted`(E_B) **do**
    if $\overrightarrow{xy} \in G_R$ **and** $G = (\mathcal{S}, E_R \cup \{\overrightarrow{xy}\})$ is acyclic **then**
        $G'_R \leftarrow (\mathcal{S}, E'_R \cup \{\overrightarrow{xy}\})$
for edge $\overrightarrow{xy} \in$ `Sorted`(E_R) **do**
    if $\overrightarrow{xy} \in G_B$ **and** $G = (\mathcal{S}, E_B \cup \{\overrightarrow{xy}\})$ is acyclic **then**
        $G'_B \leftarrow (\mathcal{S}, E'_B \cup \{\overrightarrow{xy}\})$
▷ *Construct the final rankings by converting the acyclic tournaments G'_R, G'_B to strict total orders* ◁
$R', B' \leftarrow$ empty rankings of length $l = |\mathcal{S}|$
Insert(elements of R, B not in tie-groups, R', B')
for tie-group tg **in** R **do**
    Sort(tg) by $\lambda xy : x < y \iff \overrightarrow{xy} \in E'_R$
    Insert(tg_{sorted}, R')
for tie-group tg **in** B **do**
    Sort(tg) by $\lambda xy : x < y \iff \overrightarrow{xy} \in E'_B$
    Insert(tg_{sorted}, B')
return R', B'

4.5 Edge sorting

What happens if both G_R and G_B contain both $\overrightarrow{xy}$ and $\overrightarrow{yx}$? Either one of the edges could be added, and as long as we make the same choice in both rankings, the two items x and y will be concordant. In the algorithm above, only one of the two will be added, because when we try to add the second it will create a 2-cycle, and thus be skipped. The choice of $\overrightarrow{xy}$ vs $\overrightarrow{yx}$ depends on *which one we see first*. The call to `Sort` the edges (seen in Algorithm 2) serves exactly this purpose: if we always see the edges in the same order, we will always pick the same one from a pair $\{\overrightarrow{xy}, \overrightarrow{yx}\}$.

What sorting should we use? For (unweighted) Kendall's τ it does not matter, however for τ_w *it does*. Intuitively, τ_w gives different *concordance weights* to different pairs of items. We always greedily try to make a pair concordant when we see it, so we can sort the edges by descending weight.

An experimental validation of this idea showed that not all weight functions allow Algorithm 2 to yield optimal correlation. Concretely, Algorithm 2 is optimal under the assumption that *all solutions that maximise τ also maximise concordance*. Some weight functions abide by this constraint, two notably useful examples are $w(i, j) = 1$ (unweighted Kendall's τ), and $w(i, j) = 1/\max(R_i, R_j)$ (τ_{AP} by Yilmaz et al.)[13]. Others violate this constraint, most notably $w(i, j) = 1/((R_i + 1)(R_j + 1))$ (Vigna's τ_h).

This is a fundamental limitation of the assumption we used to solve the problem in polynomial time: that maximising concordance also maximises correlation. We do not reject the possibility that the optimisation problem is tractable even in the general case, but we do not prove so in this paper. Nevertheless, even for weights where reaching maximum correlation requires less than maximum concordance, we can examine the error by using a brute-force solver to find the optimal solution. Figure 5 shows a comparison between the optimal solution and the one computed by Algorithm 2. The results seem to suggest that even though maximum (or minimum) correlation is not achieved by the permutation of ties that maximise (or minimise) concordance, the two solutions tend to be close.

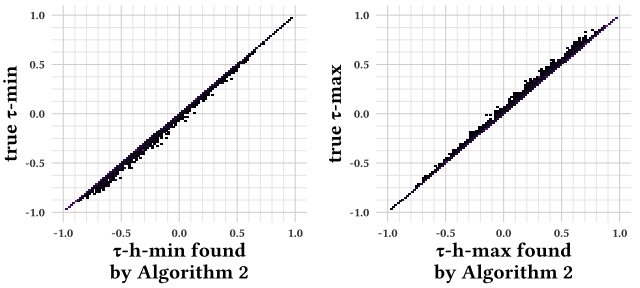


Figure 5: Heat maps comparing the true τ_h^{min} and τ_h^{max}, as computed by a brute-force solver, to the ones found by Algorithm 2, on 50 000 ranking pairs from a synthetic dataset.

4.6 Minimisation case

Minimising concorance is equivalent to maximising discordance. In our graphs, this means that for any edge $\overrightarrow{xy}$ in one graph, we ideally want the edge $\overrightarrow{yx}$ in the other graph. Moreover, when the choice between $\overrightarrow{xy}$ and $\overrightarrow{yx}$ is arbitrary, we should always pick discordant pairs. The construction of G'_R, G'_B is shown in Algorithm 3.

Algorithm 3 Edge sorting for minimising concordance

for edge $\overrightarrow{xy} \in$ `Sorted`(E_B) **do**
    if $\overrightarrow{yx} \in G_R$ **and** $G = (\mathcal{S}, E_R \cup \{\overrightarrow{yx}\})$ is acyclic **then**
        $G'_R \leftarrow (\mathcal{S}, E'_R \cup \{\overrightarrow{yx}\})$
for edge $\overrightarrow{xy} \in$ `Inverse(Sorted(`E_R`))` **do**
    if $\overrightarrow{yx} \in G_B$ **and** $G = (\mathcal{S}, E_B \cup \{\overrightarrow{yx}\})$ is acyclic **then**
        $G'_B \leftarrow (\mathcal{S}, E'_B \cup \{\overrightarrow{yx}\})$

[13] Since we have not formalised the restriction on w, we compared the bounds found by Algorithm 2 to the brute-force solver on a synthetic dataset of 250k ranking pairs, with a number of permutations of at most 250 million. On τ_{AP}, none of the test cases showed any difference between the computed and reference solutions.

4.7 Implementation

An implementation of this algorithm, generalised for τ_w, can be found on https://github.com/andtsa/bsc_thesis.

4.8 Proof of optimality

We have proven the equivalence of the problem of finding the the permutation of ties that minimises (or maximises) τ, and the problem of minimising (or maximising) the set intersection of two acyclic tournament subgraphs of the partial-order-graph representations. The only missing piece in the optimality of this algorithm is Theorem 5: *picking an edge that increases concordance without creating a cycle is always optimal.* This is fundamentaly a greedy algorithm, and we will now prove that this greedy algorithm can produce a solution at least as good as any optimal solution.

Proof. We can prove this with an exchange argument. Start with the two partial total order graphs

$$G_R = (\mathcal{S}, E_R),\ G_B = (\mathcal{S}, E_B)$$

(from section 4.1) and the two acyclic graphs

$$G'_R = (\mathcal{S}, E'_R), G'_B = (\mathcal{S}, E'_B)$$

(as explained in Section 4.3). Let $G^*_R = (\mathcal{S}, E^*)$ be any acyclic tournament subgraph of G_R such that

$$|E^* \cap E_B| = \max |E' \cap E_B| \text{ over all acyclic } E' \subseteq E_R$$

Meaning G^* is an optimal solution.

Suppose the greedy algorithm is considering an edge $e = \overrightarrow{xy} \in E_B \cap E_R$, where adding e to E'_R does not create a cycle.

Case 1. $e \in E^*$ Trivial, the optimal solution already contains e

Case 2. $e \notin E^*$ We will show that including $\overrightarrow{xy}$ in E'_R yields a solution no worse than E^*.

We know that G^* is an acyclic and thus transitive tournament, and since $\overrightarrow{xy} \notin E^*$, then $\overrightarrow{yx} \in E^*$. Since $\overrightarrow{xy} \in E_B$, including $\overrightarrow{xy}$ in either solution E'_R or E^* will increase concordance by 1: either because $\overrightarrow{yx} \notin E_B$, which means in the final orders R', B', after including $\overrightarrow{xy}, x \prec_{R'} y$ and $x \prec_{B'} y$. Otherwise, if both $\overrightarrow{xy} \in E_B$ and $\overrightarrow{yx} \in E_B$, concordance will increase because the construction of G'_B is guaranteed to make the same choice as the greedy algorithm is making now (see Section 4.5 for explanation).

Now we need to prove that we can impose the greedy choice on the optimal solution, without losing optimality or validity in the solution. G^* being a transitive tournament also means that reversing $\overrightarrow{yx}$ (into $\overrightarrow{xy}$) in E^* cannot create a cycle, *unless* there was already a directed path $y \rightsquigarrow x$. If $(y \rightsquigarrow x) \notin E^*$ then we are finished, since we have shown that a greedy solution is valid and at least as good as an optimal one (an optimal solution can be converted to the greedy solution without decreasing concordance). Suppose we have the other case, where $(y \rightsquigarrow x) \in E^*$, and adding e to $E^* \setminus \{\overrightarrow{yx}\}$ closes a cycle. We call C the directed cycle $x \to y \to \ldots \to x$. We know that C *cannot* lie entirely inside E'_R (because we only add edges to E'_R if they do not create a cycle), so there is a non-empty set of edges F such that:

$$F \subseteq C, F \subseteq (E^* \setminus \{\overrightarrow{yx}\} \cup \{\overrightarrow{xy}\}), F \cap E'_R = \emptyset$$

Because G^* is a transitive tournament, $\mathcal{S}$ has a unique topological sort according to E^*. We can always pick an edge $f \in F$ that connects two topologically adjacent vertices, and reverse it, without creating a cycle. If $\overrightarrow{yx} \notin E_B$ then we can flip f, losing 1 concordance, and maintaining an optimal solution, since we gained 1 by adding $\overrightarrow{xy}$. If $\overrightarrow{yx} \in E_B$, then either $\overrightarrow{xy} \in E'_B$ or $\overrightarrow{yx} \in E'_B$. For the second case to happen, during construction of E'_B, adding $\overrightarrow{xy}$ would have caused a cycle (since we know we tried $\overrightarrow{xy}$ before $\overrightarrow{yx}$). This is impossible, since we used the same prefix of edges in constructing E'_B as our initial construction of E'_R used (the sorting is the same), and we already know that $\overrightarrow{xy}$ does not close a cycle in E'_R.

Regardless, in any combination of the above cases, we have that $E^{*\prime} = E^* \setminus \{\overrightarrow{yx}, f\} \cup \{\overrightarrow{xy}, \bar{f}\}$ is a valid solution, with concordance no lower than E^*. Since at every step of the greedy construction we are able to get to a solution at least as good as an optimal solution, the greedy algorithm for selecting edges is optimal.

■

5 Quantifying uncertainty

For rankings with ties, in addition to the τ or τ_w correlation (using the τ^a or τ^b variants described by Kendall [9], or τ_w with ties from Vigna [15]) we can now compute the highest and lowest possible correlation values, in all scenarios where the ranking algorithms made a decision. As we described in Section 2, these two values together show the uncertainty that the correlation value does not show on its own.

Our hypothesis, largely motivated by the similar study on RBO [3], is that *in many real-world rankings, ties contribute an uncertainty large enough that it cannot be overlooked.* Logically, the null hypothesis is that *for most real-world rankings, the uncertainty is close enough to zero* (meaning the bounds are tight around one of the τ variants) *that we can confidently assume it is negligible.*

To support or contradict our hypothesis, we examined the computed τ, τ^{max}, τ^{min} values, across two datasets of rankings: one dataset we synthesised ourselves, and the dataset of TREC Web ad-hoc systems, from years 2010 to 2014. The intention with the first dataset is to examine rankings in the general case, without any domain-specific features, while the latter dataset represents real-world data that contains ties caused by uncertainty. The Text REtrieval Conference (TREC) datasets can be retrieved from https://trec.nist.gov/data.html. In Section 5.1 we will explain how we generated the synthetic data, and present the bounds calculations on it. In Section 5.2 we will perform the same analysis on the TREC datasets.

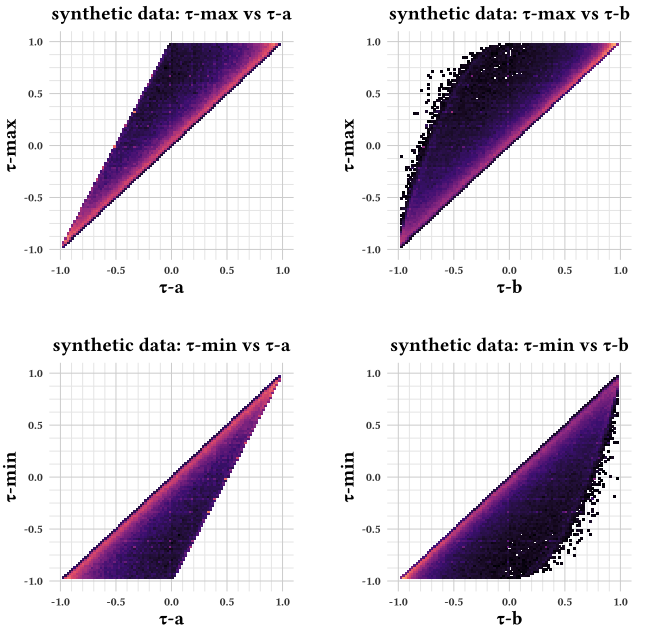


Figure 6: Heat maps comparing the values of τ variants τ^a, τ^b to the upper and lower bounds for the same rankings. Plots were created by computing the correlation and bounds on 250 000 ranking pairs from the synthetic dataset.

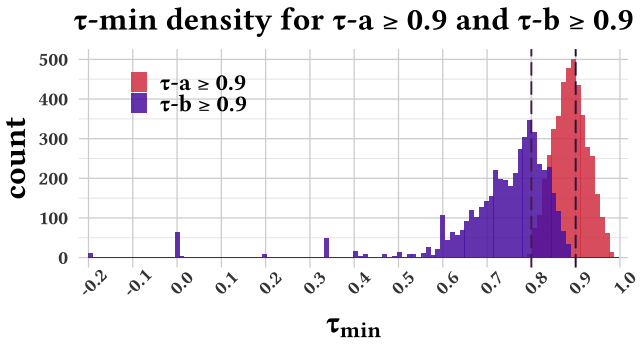
5.1 Synthetic data

As already briefly stated, the goal of the synthetic data generation is to create a diverse set of rankings, whose randomness is not bound by any properties of the items they are ranking. This gives a basis to claim that the findings will be applicable to rankings in any domain of science.

To generate the rankings we used the method described by Corsi and Urbano [4] (see https://github.com/julian-urbano/sigir2024-rbo) for simulating ranking pairs given a target τ value, lenghts, and the fraction of items that should be in ties. 250 000 rankings were simulated along a uniform distribution for `frac_ties` ($[0, 1]$), with a uniform distribution ($[-1, 1]$) for τ, and with lengths in $[3, 150]$.

In Figure 6 a comparison of the τ^a and τ^b values against the τ^{min} and τ^{max} values for the same ranking pairs is shown. If ties where to have little impact on the possible values for τ, we would expect the bounds τ^{min} and τ^{max} to be centred around the main diagonal, indicating that they differ little from the τ^a or τ^b approximations. This is clearly not the case, which supports our hypothesis that neither τ^a nor τ^b alone capture the uncertainty induces by ties.

Additionally, we can examine the cases most prevalent in IR Research: those with $\tau \geq 0.9$[14]. From the initial set of 250 000 cases, the distributions for both τ^a and τ^b are visualised in Figure 7. While the results of τ^a seem to show that it gives more accurate estimates than τ^b, they are very unlikely to be seen in practice, since many major statistical libraries (SciPy [11], (base) R [10], Pandas–which delegates to SciPy) only implement the τ^b variant for Kendall's τ. These results indicate that along the uniform sampling of rankings, $\frac{7242}{9100} \approx 81.5\%$ of use cases of τ^b yield potentially false positive results. Finally, we investigated whether there was any unifying characteristic among all the rankings with low uncertainty (τ^{min} being very close to τ^a or τ^b). Other than *not containing ties at all*, we did not identify any such characteristic, which we deem to be a strong argument for the descriptive importance of τ^{min} and τ^{max}.



	total	$\tau^{\min} < 0.9$	$\tau^{\min} < 0.85$	$\tau^{\min} < 0.8$
$\tau^a \geq 0.9$	4413	2737	775	0
$\tau^b \geq 0.9$	9100	7424	5163	3238

Figure 7: Histogram of $\tau^{\min}$ density, for synthetic data with $\tau^a \geq 0.9$ or $\tau^b \geq 0.9$ (top). Count of data points at various $\tau^{\min}$ thresholds (bottom). If using solely τ^b to draw conclusions about correlation, as much as $\frac{7424}{9100} \approx 81.5\%$ of ranking paris yield potentially false positive results.

5.2 TREC data

The comparison of document rankings is better suited to similarity measures for non-conjoint rankings such as RBO [16]. In the Text Retrieval Conference, after each system is evaluated on a certain corpus of documents, the systems themselves are ranked based on different metrics, and on their performance across different topics. These rankings are conjoint, and as such constitute a real use case for τ correlation. We put together a dataset with such rankings, extracted from the results of the 2010-2014 editions of the conference. In Figure 8 we can see the same behaviour as in Figure 6, with the spread away from the centre even more pronounced, since rankings without (or with very few) ties did not occur in the data.

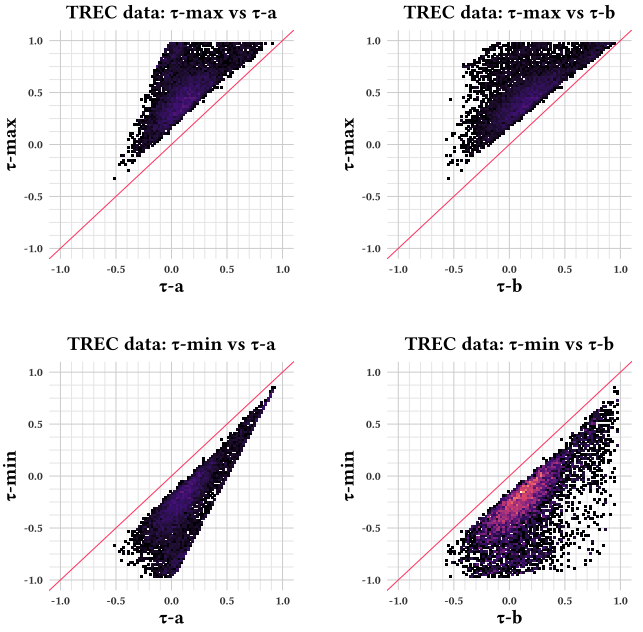


Figure 8: Heat maps comparing the values of τ variants τ^a,τ^b to the upper and lower bounds for the same rankings. Plots were created by computing the correlation and bounds on 7361 ranking pairs from the TREC dataset.

6 Responsible research

As with any scientific work, it is paramount that the claims made in this paper can be verified by the reader. We will now go over the steps we took to ensure this is the case.

In addition to the analysis of the algorithm here, we have published a git repository[15] containing all the source code used for this research. This includes implementations of the described algorithms, as well as all the tooling and instructions necessary to reproduce the findings. The TREC datasets were provided by the thesis supervisor Julián Urbano, but can be requested from NIST on their website, with access provided to anyone who intends to use the datasets solely for research.

Since we have shown the statistical relevance of these bounds, any future research that relies on Kendall's τ (or variants such as τ_{AP} by Yilmaz et al.) ought to take this research into consideration, to ensure any decisions based on correlation values are made with knowledge of the potential error due to ties. We argue that not only does this research have no discernible ethical downsides, but that including the calculations of τ^{min} and τ^{max} in any future work can only aid in limiting the possibility of false positive results.

7 Future work

We suggest future work could explore the missing constraints on the weight function w, as well as an algorithm (or extension of the algorithms shown here) that can always arrive at the optimal solution, for all w of τ_w. Additionally, the proposed algorithm could be adapted to produce a *distribution* of correlation values across the permutation of ties, providing a more informative description of the effect of ties. Finally, future work could explore the possibility of introducing the algorithms proposed here to popular software packages used for scientific computing. This would not only encourage the use of the τ^{min}, τ^{max} bounds, but also facilitate their computation by the broader scientific community.

[14] 0.9 being a common lower threshold of correlation, from which positive results can be drawn.

[15] https://github.com/andtsa/bsc_thesis

8 Conclusion

In this research we examined the effect on ties on τ_w correlation, in the case where ties represent an uncertainty in the ranking. Having found that the uncertainty can often be high enough to influence or even alter decisions made based on the raw correlation value, given from one of the variants τ^a or τ^b, we introduced the uncertainty variants τ^{min} and τ^{max}. We proposed an algorithm for computing τ^{min} and τ^{max} in quadratic time, and presented a proof of its correctness, under certain restrictions on the weight function.

By pairing each point estimate of Kendall's τ with its corresponding $[\tau^{min}, \tau^{max}]$ interval, researchers can gain a clearer understanding of the true strength and reliability of rank correlations under ties. Our empirical results, across both synthetic and TREC datasets, demonstrate that these intervals are not merely theoretical artifacts, but practical tools that can enable more informed decisions. We anticipate that embedding such uncertainty quantification directly into statistics software packages can encourage widespread adoption, and more importantly, encourage researchers to question how the tools and metrics they are using reflect the properties of the rankings in their field.

References

[1] Hans L. Bodlaender, Fedor V. Fomin, Arie M. C. A. Koster, Dieter Kratsch, and Dimitrios M. Thilikos. 2011. A Note on Exact Algorithms for Vertex Ordering Problems on Graphs. 50, 3 (2011), 420–432. doi:10.1007/s00224-011-9312-0

[2] A. Bravais. 1846. Analyse mathématique sur les probabilités des erreurs de situation d'un point. 9 (1846), 255–332. http://gallica.bnf.fr/ark:/12148/bpt6k5516293p/f261

[3] Matteo Corsi and Julián Urbano. 2024. How do Ties Affect the Uncertainty in Rank-Biased Overlap?. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region* (2024-12) *(SIGIR-AP 2024)*. Association for Computing Machinery, 125–134. doi:10.1145/3673791.3698422

[4] Matteo Corsi and Julián Urbano. 2024. The Treatment of Ties in Rank-Biased Overlap. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2024-07) *(SIGIR 2024)*. Association for Computing Machinery, 251–260. doi:10.1145/3626772.3657700

[5] Brian A. Davey and Hilary A. Priestley. 1990. *Introduction to lattices and order*. Cambridge University Press. http://www.worldcat.org/search?qt=worldcat_org_all&q=0521367662

[6] Mikołaj Jakub Gazeel. 2025. The Definition of a New Tie Variant for Conjoint Rank Similarity Measures.

[7] M. G. Kendall. 1938. A New Measure of Rank Correlation. 30, 1–2 (1938), 81–93. doi:10.1093/biomet/30.1-2.81

[8] Maurice G. Kendall. 1945. *Rank Correlation Methods.*

[9] M. G. Kendall. 1945. The Treatment of Ties in Ranking Problems. 33, 3 (1945), 239–251. doi:10.1093/biomet/33.3.239

[10] R Core Team. 2023. *Correlation [cor] in base R.* https://stat.ethz.ch/R-manual/R-devel/library/stats/html/cor.html R 4.3.1 Manual, accessed 2025-06-20.

[11] SciPy Developers. 2024. *scipy.stats.kendalltau.* https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kendalltau.html SciPy v1.15.3 Documentation, accessed 2025-06-20.

[12] Grace S. Shieh. 1998. A weighted Kendall's tau statistic. 39, 1 (1998), 17–24. doi:10.1016/S0167-7152(98)00006-6

[13] C. Spearman. 1904. The Proof and Measurement of Association between Two Things. 15, 1 (1904), 72–101. http://www.jstor.org/stable/1412159

[14] A. Stuart. 1953. The Estimation and Comparison of Strengths of Association in Contingency Tables. 40, 1/2 (1953), 105–110. http://www.jstor.org/stable/2333101

[15] Sebastiano Vigna. 2014. A Weighted Correlation Index for Rankings with Ties. abs/1404.3325 (2014). arXiv:1404.3325 http://arxiv.org/abs/1404.3325

[16] William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. 28, 4 (2010), 1–38. doi:10.1145/1852102.1852106

[17] Emine Yilmaz, Javed A. Aslam, and Stephen Robertson. 2008. A new rank correlation coefficient for information retrieval. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore, 2008) *(SIGIR '08)*. Association for Computing Machinery, 587–594. doi:10.1145/1390334.1390435

Appendices

A Variants of Kendall's τ

a) For the variant τ^a, simply define $sgn(0) = 0$, and compute using the same formula. This leads to the probabilistic interpretation of *taking the average correlation across all permutations*. As soon as two items are tied, the numerator of equation 4 can never be equal to $n(n-1)/2$, which means that if there are ties, τ^a can never be equal to 1 or -1. τ^b amends this by adjusting the denominator based on the number of ties.

b) For the variant τ^b, define $sgn(0) = 0$, and additionally change the denominator of equation 4 to

$$\frac{\sqrt{\sum_{\{i,j\}} |sgn(r_i - r_j)| + |sgn(b_i - b_j)|}}{2}$$

c) For the variant τ^c, let $n_0 = n(n-1)/2$, n_c = #concordant pairs, n_d = #discordant pairs, and let r = #distinct r_i, c = #distinct b_i, $m = \min(r, c)$. Then Stuart–Kendall's τ^c is defined by

$$\tau^c = \frac{2\,(n_c - n_d)}{n^2\,\frac{m-1}{m}} = \tau^a \times \frac{n-1}{n}\,\frac{m}{m-1}, \tag{20}$$

so that even when $r \neq c$ the range of τ^c is still $[-1, 1]$.

d) Vigna showed that we can compute τ_w in the presence of ties [15] as:

$$\tau_w = \frac{\sum_{\{x,y\}} c_{R,B}(x,y) w(x,y)}{\sqrt{\sum_{\{x,y\}} |c_{R,R}(x,y) w(x,y)|}\sqrt{\sum_{\{x,y\}} |c_{B,B}(x,y) w(x,y)|}} \tag{21}$$

B Remaining Proofs

Theorems 1 and 2 can be considered general knowledge, and are by no means a novel contribution of this paper. Nevertheless, the proofs are ours, and we believe can provide some useful intuition for the curious reader. More related information can be found on https://en.wikipedia.org/wiki/Tournament_(graph_theory) as well as "Introduction to lattices and order" by Davey and Priestley [5].

Theorem 1. Transitive tournaments are equivalent to strict total orders

Proof. For a transitive tournament T, we can fulfill all the properties of a strict total order (Definition 2) as follows:

(1) Irreflexive: $\neg(a < a)$: no self-loops, follows from our definition of tournaments (Definition 4).
(2) Asymmetric: $a < b \implies \neg(b < a)$: no 2-cycles, since T is a tournament, there can only be one edge connecting a and b.
(3) Transitive: $(a < b \wedge b < c) \implies a < c$: follows from the fact that T is a transitive tournament.
(4) Total: $a \neq b \implies (a < b \vee b < a)$: follows from definition of tournament.

To prove equivalence, the opposite must also hold: a strict total order must satisfy the definition of a transitive tournament. We can prove this by construction: if we have a STO $\prec_s$ on set $\mathcal{X}$, we construct $T = (\mathcal{X}, E)$ where

$$E = \{\overrightarrow{xy} | x, y \in \mathcal{X}, x \prec_s y\}. \tag{22}$$

Since $\prec_s$ is total, E contains an edge between every pair of vertices, which in turn implies:

$$(\forall a, b, c \in \mathcal{X} : a \prec_s b \wedge b \prec_s c \implies a \prec_s c)$$
$$\implies (\forall a, b, c \in \mathcal{X} : \overrightarrow{ab}, \overrightarrow{bc} \in E \implies \overrightarrow{ac} \in E)$$

And thus we have that $(\prec_s, \mathcal{X})$ is a transitive tournament. ■

Theorem 2. Every acyclic tournament is a transitive tournament, and every transitive tournament is acyclic.

Proof. We will prove both directions by contradiction. For the first direction (an acyclic tournament is transitive), suppose we have an acyclic tournament $T = (V, E)$, and that T is not transitive, meaning

$$\exists a, b, c \in T : \overrightarrow{ab} \in E \wedge \overrightarrow{bc} \in E \wedge \overrightarrow{ac} \notin E \tag{23}$$

From Definition 4 we know that there is always exactly one edge (of the two possible) between every two vertices, Equation 23 states that $\overrightarrow{ac} \notin E$, which impliy that $\overrightarrow{ca} \in E$. This means we have a cycle $\{\overrightarrow{ab}, \overrightarrow{bc}, \overrightarrow{ca}\}$, which is a contradiction.

The second case can be trivially proven in the same way as the first, and is left as an exercise for the reader. ■