Transductive Conformal Inference for Full Ranking

Jean-Baptiste Fermanian
IMAG, IROKO, Univ. Montpellier, Inria, CNRS,
Montpellier, France
jean-baptiste.fermanian@inria.fr

Pierre Humbert
LPSM, Sorbonne Université,
Paris, France
humbert@lpsm.paris

Gilles Blanchard
Université Paris Saclay, Institut Mathématique d'Orsay,
Orsay, France
gilles.blanchard@universite-paris-saclay.fr

Abstract

We introduce a method based on Conformal Prediction (CP) to quantify the uncertainty of full ranking algorithms. We focus on a specific scenario where $n + m$ items are to be ranked by some "black box" algorithm. It is assumed that the relative (ground truth) ranking of n of them is known. The objective is then to quantify the error made by the algorithm on the ranks of the m new items among the total $(n+m)$. In such a setting, the true ranks of the n original items in the total $(n + m)$ depend on the (unknown) true ranks of the m new ones. Consequently, we have no direct access to a calibration set to apply a classical CP method. To address this challenge, we propose to construct distribution-free bounds of the unknown conformity scores using recent results on the distribution of conformal p-values. Using these scores upper bounds, we provide valid prediction sets for the rank of any item. We also control the false coverage proportion, a crucial quantity when dealing with multiple prediction sets. Finally, we empirically show on both synthetic and real data the efficiency of our CP method for state-of-the-art ranking algorithms such as RankNet or LambdaMart.

1 Introduction

Ranking is a fundamental problem in machine learning where the objective is to sort some items, such as documents or products, by their relevance to a query or user profile. It has a wide range of applications, ranging from document retrieval [Cao et al., 2006], collaborative filtering [Liu and Yang, 2008], sentiment analysis [Liu et al., 2017], and product rating. These applications are grouped under the term *preference learning* [Fürnkranz et al., 2008]. While there exist a lot of ranking algorithms in the literature (see e.g. Liu et al. [2009] for a review), the quantification of the uncertainty in their rank predictions using conformal prediction is relatively new [Angelopoulos et al., 2023a, Xu et al., 2024]. Indeed, so far, the design of ranking algorithms has mainly focused on the training phase and their evaluations regarding the uncertainty of their predicted ranking, is not often studied. With the increasing popularity of ranking methods (especially in the learning-to-rank literature Liu et al., 2009), this evaluation is however a mandatory step to deploy them for real-world applications.

Our main objective is to quantify the uncertainty of ranking algorithms by constructing distribution free prediction set for the ranks of each items. Formally, given $n + m$ items with an underlying unknown total order, we want to construct prediction sets that contain the true rank of each item with high probability. We consider a particular scenario where the first n items have known relative rankings, but their positions within the whole $n + m$ items are unknown. Furthermore, the remaining m items have no known ranking information.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

This scenario occurs for many problems. In a matchmaking problem for instance [Herbrich et al., 2006, Alman and McKay, 2017, Minka et al., 2018], the m new items are new players entering the game that we want to rank among the previous players to organize matches between players with similar skills. In product rating, the m new items can be new products such as movies that we want to rank among those already known by a particular user for which we have his feedback. In these cases, we can assume that we have access to the exact relative rank of the n first items. We also consider the case where we want to combine an expensive ranking algorithm with a cheaper but less reliable one (based, for example, on a smaller architecture). In this situation, we want to estimate the reliability of the cheaper algorithm compared to the costly one considered as a proxy truth. We can therefore randomly select n items, rank them using the more efficient algorithm, and then use these rankings to quantify the uncertainty of the less efficient algorithm when it ranks the remaining m items.

Formally, for each item $i \in [\![n+m]\!]$, we want to construct a *marginally valid* set which contains, with high probability, its unknown rank R_i inside the $n+m$ items:

$$\mathbb{P}\Big(R_i \in \widehat{\mathcal{C}}_i\Big) \geqslant 1-\alpha \; , \tag{1}$$

where $\alpha \in (0,1)$ is a desired miscoverage level. In addition, as we construct prediction sets for multiple points, we also want to control the false coverage proportion, which is the average number of mispredictions. More precisely, we want to construct m prediction sets $(\widehat{\mathcal{C}}_i)_{i=1}^m$ satisfying for some small $\beta > 0$:

$$\mathbb{P}\left(\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\{R_i \notin \widehat{\mathcal{C}}_i\} \leqslant \alpha\right) \geqslant 1-\beta \; . \tag{2}$$

There exist several methods to construct marginally valid sets in regression or classification settings, based on the so-called Conformal Prediction (CP) framework [Papadopoulos et al., 2002, Vovk et al., 2005, Romano et al., 2019], but these do not directly apply in this specific setting of ranking. Indeed, the problem of ranking can be seen as a regression problem where the outcome, the rank, depends on all the points observed, or as a classification problem with a moving number of classes. An important difference with a classical CP framework is that a calibration set is not directly accessible, as knowing the relative rank of the n first items is not sufficient to know their ranks within the whole sample.

Contributions. This work constructs marginally valid prediction sets, satisfying Eq. (1), for the ranks of $n+m$ items. We assume already having a ranking algorithm $\mathcal{A}$ (considered as a "black box") and knowing the exact relative ranks of n items. These n items will be used to quantify the uncertainty of the algorithm $\mathcal{A}$. Our main contribution is to propose a general method for this calibration phase. The difficulty is that we cannot directly quantify the algorithm's error on the calibration set, as the true ranks are only known through the relative ranks. We then provide computable bounds of these ranks, supported by theoretical arguments, to construct valid prediction sets for which we also control the false coverage proportion. These findings provide a practical solution to the challenges of constructing valid prediction sets in ranking problems, where traditional CP methods face limitations due to the interdependence of ranks.

2 Background

2.1 Split conformal prediction

Conformal Prediction (CP) is a framework introduced by Vovk et al. [2005] to construct distribution-free prediction sets quantifying the uncertainty in the predictions of an algorithm. Formally, given a calibration set $\{(X_i, Y_i)\}_{i\in[\![n]\!]}$ of points in $\mathcal{X} \times \mathcal{Y}$ and an algorithm $\mathcal{A}$, a CP method constructs for a new point $X_{n+1} \in \mathcal{X}$ a set $\widehat{\mathcal{C}}$ containing the unobserved outcome $Y_{n+1} \in \mathcal{Y}$ with high probability:

$$\mathbb{P}\Big(Y_{n+1} \in \widehat{\mathcal{C}}(X_{n+1})\Big) \geqslant 1-\alpha \, . \tag{3}$$

In classification, $\mathcal{Y}$ is the set of classes and $\mathcal{A}$ a classifier and in regression, $\mathcal{Y} = \mathbb{R}$ and $\mathcal{A}$ is a regressor. The idea behind CP methods is to first build non-conformity scores $V_i := s(X_i, Y_i)$ on the calibration set, where $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, is a score function which is intended to quantify the error of the algorithm at a given point. Many score functions have been considered to catch different type of information, see e.g. Angelopoulos et al. [2023b] for a review. In regression, a common choice is to use the absolute residual $s(x, y) = |y - \mathcal{A}(x)|$ where the algorithm $\mathcal{A}$ can either be trained on an independent sample (split conformal prediction, Papadopoulos et al., 2002) or retrained on a subset of points (full conformal prediction Vovk et al., 2005). In this paper, we will focus on the case where an already trained algorithm is provided, which corresponds to the case of split CP. Then, for this score function and some integer k, the prediction set is

$$\widehat{\mathcal{C}}_k(x) := \left\{ y \in \mathcal{Y} \, : \, s(x, y) \leqslant V_{(k)} \right\} , \tag{4}$$

where $V_{(k)}$ is the k-th smallest score of $V_1, \ldots, V_n, \infty$. This set is marginally valid under mild assumptions as presented in the following theorem.

Theorem 2.1 (Vovk et al., 2005, Lei et al., 2018)**.** *If the scores $V_1, \ldots, V_{n+1}$ are exchangeable, then for any $\alpha \in (0, 1)$ and $k = \lceil (1 - \alpha)(n + 1) \rceil$ the set* (4) *returned by the split CP method satisfies:*

$$\mathbb{P}\Big(Y_{n+1} \in \widehat{\mathcal{C}}_k(X_{n+1}) \Big) \geqslant 1 - \alpha \ .$$

We refer to Vovk et al. [2005], Angelopoulos et al. [2023b] and Fontana et al. [2023] for in-depth presentations of CP. See also Manokhin [2024] for a curated list of CP papers.

2.2 False coverage proportion

Some recent results have considered the transductive setting [Vovk, 2013] where conformal prediction sets are constructed for $m \geqslant 2$ test points $\{(X_{n+i}, Y_{n+i})\}_{i=1}^m$. In this case, while maintaining the coverage guarantee (3), a usual goal is to control the False Coverage Proportion (FCP) defined as:

$$\text{FCP} := \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}\Big\{ Y_{n+i} \notin \widehat{\mathcal{C}}(X_{n+i}) \Big\} , \tag{5}$$

where $\widehat{\mathcal{C}}(X_{n+i})$ are some prediction sets. They can be constructed as in Eq. (4) for instance. In words, the FCP is the proportion of observations in the test set that fall outside the constructed prediction sets.

If the prediction sets are of the form (4), under the assumptions of Theorem 2.1 we have $\mathbb{E}[\text{FCP}] \leqslant \alpha$. Furthermore, the exact distribution of the FCP is known [Marques F., 2024, Huang et al., 2024]. Going further, Gazin et al. [2024] studied the full joint distribution of the insertion ranks of test scores into calibration scores, leading under the same assumptions to a uniform control of the FCP. The asymptotic behaviour of the FCP in the form of a functional CLT has also been studied recently by Gazin [2024]. With all these theoretical results, it is possible to control in probability the FCP by adjusting the coverage of the prediction sets.

2.3 Related work in ranking

Given a set of items the goal of ranking is to infer a particular ordering over these items. There exist a multitude of settings and approaches to solve the ranking problem.

In the learning-to-rank literature, and more specifically, in the pairwise approach, the ranking task is formalized as a classification of pairs of items into two classes: a tuple is correctly ranked or incorrectly ranked. Herbrich et al. [1999] proposed an algorithm based on Support Vector Machine (SVM) called RankSVM to solve this classification problem. Alternative techniques can be used such as boosting with RankBoost [Freund et al., 2003], Neural Network with RankNet [Burges et al., 2005], random forest with LambdaRank [Burges et al., 2006] and LambdaMart [Wu et al., 2010] (see Burges, 2010, Qin et al., 2021 for an overview). Other methods based on the pointwise or listwise approaches are also widely study [Cao et al., 2007, Liu et al., 2009, Li, 2011].

Another line of research solve the ranking problem through binary comparisons of items. The different settings considered allow these binary comparisons to be measured either completely at random [Wauthier et al., 2013], actively [Ailon, 2012, Heckel et al., 2019], or several times [Negahban et al., 2012, Feige et al., 1994]. They can also assume that all the possible comparisons between the items are known but up to some noise [Braverman and Mossel, 2007, 2009]. A significant body of literature also focuses on the problem of uncertainty quantification in ranking problems under the Bradley-Terry-Luce model [Bradley and Terry, 1952, Luce et al., 1959]. For instance, Liu et al.

[2023] consider the BTL model and infer its general ranking properties. Gao et al. [2023] and Fan et al. [2025] study the maximum likelihood estimator and the spectral estimator to estimate the parameters of the BTL model, enabling them to construct confidence intervals for individual ranks.

However, the quantification of uncertainty of these ranking algorithms has been little studied in a CP framework. Recently, Angelopoulos et al. [2023a] have developed a CP method for quantifying the uncertainty of learning-to-rank algorithms specially tailored for recommendation systems. Xu et al. [2024] also proposed to use CP, but for the ranked retrieval task. While related to the present work, they do not consider the same setting.

3 Conformal prediction for ranking algorithms

We now describe our setting, as well as our methodology to provide valid prediction sets.

Notation: For a point y in $\mathbb{R}$ and a finite subset $\mathcal{D} \subset \mathbb{R}$, the rank of y in $\mathcal{D}$ is $\mathtt{R}(y, \mathcal{D}) := \sum_{z \in \mathcal{D}} \mathbf{1}\{y \geqslant z\}$. For an index $1 \leqslant i \leqslant |\mathcal{D}|$, if $\mathcal{D}$ has no ties, we denote the point of $\mathcal{D}$ of rank i by $\mathtt{R}^{-1}(i, \mathcal{D}) := \sum_{z \in \mathcal{D}} z \mathbf{1}\{\mathtt{R}(z, \mathcal{D}) = i\}$. A multiset (or bag) will be denoted $\lbag \cdot \rbag$ and is an unordered set with potentially repeated elements.

3.1 Setting

Let us consider $n + m$ items, each associated with a pair $(X_i, Y_i) \in \mathcal{X} \times \mathbb{R}$. We suppose that these items are divided into two sets: a calibration set of size n, $\{(X_i, Y_i)\}_{i \in \llbracket n \rrbracket}$ and a test set of size m, $\{(X_{n+i}, Y_{n+i})\}_{i \in \llbracket m \rrbracket}$. For any item i, X_i is, for instance, a vector of features, while Y_i allows us to properly define its true rank within each subset or the entire set through the following variables:

$$R_i^c = \mathtt{R}\big(Y_i, \lbag Y_j \rbag_{j \in \llbracket n \rrbracket}\big),\ R_i^t = \mathtt{R}\big(Y_i, \lbag Y_{n+j} \rbag_{j \in \llbracket m \rrbracket}\big),\ R_i^{c+t} = \mathtt{R}\big(Y_i, \lbag Y_j \rbag_{j \in \llbracket n+m \rrbracket}\big) = R_i^c + R_i^t\ .$$

We assume throughout the paper that:

Assumption 3.1. The vector $(Y_i)_{i \in \llbracket n+m \rrbracket}$ is *exchangeable* and has *no ties*. In particular, this implies that there is a *total order* between the $n + m$ items.

The values $(Y_i)_i$ can be interpreted as an *unobservable underlying truth*. In our setting, we insist that only the features $(X_i)_{i \in \llbracket n+m \rrbracket}$ and the true ranks of the calibration points $(R_i^c)_{i \in \llbracket n \rrbracket}$ are observed. For example, they can model the intrinsic skill of some players only observable by comparing them with each other. Assumption 1 is needed to model correctly our problem of full ranking. The following assumption is essential to be able to construct conformal prediction sets for the ranks.

Assumption 3.2. The vector $(X_i, R_i^{c+t})_{i \in \llbracket n+m \rrbracket}$ is exchangeable.

To estimate the true ranks $R_1^{c+t}, \ldots, R_{n+m}^{c+t}$, we assume that we have access to an algorithm $\mathcal{A} : \mathcal{X} \times \mathcal{X}^{n+m} \to \mathbb{K}$ which is intended to predict the rank of a point inside a set of points. Its inputs are the target item and the multiset of items among which it seeks to sort it. For clarity, when there is no confusion, we will sometimes omit the dependence in the multiset of items: $\mathcal{A}(x) := \mathcal{A}(x, \lbag x_\ell \rbag_\ell)$. This algorithm can predict for each point either a *rank* or a *value* from which a rank can be deduced. For these two situations, we propose two conformity score functions which quantify the error made by $\mathcal{A}$ in its ranking:

(RA) setting: $\mathbb{K} = \mathbb{N}$, the rank is directly predicted by the algorithm $\mathcal{A}$: $\widehat{R}_i^{c+t} := \mathcal{A}\big(X_i, \lbag X_j \rbag_{j \in \llbracket n+m \rrbracket}\big)$. In this situation, we can consider the classical residual scores, i.e., for $x, (x_j)_j$ in $\mathcal{X}$ and $r \in \mathbb{N}$:

$$s^{\mathbf{RA}}(x, r, \lbag x_j \rbag_j) := |r - \mathcal{A}(x, \lbag x_j \rbag_j)|\ .$$

This is simply the absolute difference between r and the predicted rank of x inside $(x_j)_j$.

(VA) setting: $\mathbb{K} = \mathbb{R}$, the predicted rank is deduced from values constructed by the algorithm $\mathcal{A}$:

$$\widehat{R}_i^{c+t} := \mathtt{R}\Big(\mathcal{A}\big(X_i, \lbag X_\ell \rbag_\ell\big), \Big\{\mathcal{A}\big(X_j, \lbag X_\ell \rbag_\ell\big)\Big\}_{j \in \llbracket n+m \rrbracket}\Big).$$

Here, because $\mathcal{A}(x)$ is a real value, we consider a more refined score function than $s^{\mathbf{RA}}$, for $x, (x_j)_j$ in $\mathcal{X}$ and $r \in \mathbb{N}^*$:

$$s^{\mathbf{VA}}(x, r, \lbag x_j \rbag_j) := \left| \mathtt{R}^{-1}\Big(r, \lbag \mathcal{A}(x_j) \rbag_j\Big) - \mathcal{A}(x) \right| .$$

This score quantifies the distance between the value attributed by the algorithm to the point x and the value it should have had to be at rank r. We restrict ourselves to these two scores for our experiments, but our methodology and theoretical results are more generally valid for any score function $s(x, r, \lbag x_i \rbag_i)$.

Remark 3.3. The fact that the algorithm depends on the multiset rather than an $(n+m)$-tuple is important, as it implies that the algorithm treats the points exchangeably. With Assumption 3.2, the vector $(\mathcal{A}(X_i))_{i \in \llbracket n+m \rrbracket}$ is thus exchangeable.

Remark 3.4. By considering the predicted ranks as scores, we see that (**RA**) is a sub-case of (**VA**). Moreover, as in $(\mathbf{RA})$, $\mathtt{R}^{-1}(r, \lbag \mathcal{A}(x_i) \rbag_i)_i = r$, $s^{\mathbf{VA}}$ reduces to the residual function $s^{\mathbf{RA}}$.

3.2 Methodology and main results

We now introduce our main theoretical results to construct prediction sets for the ranks $R_1^{c+t}, \dots, R_{n+m}^{c+t}$. Recall that our objective is to build for $j \in \llbracket n+m \rrbracket$ a marginally valid set $\widehat{\mathcal{C}}_j$ for R_j^{c+t}, i.e. satisfying Eq. (1) for a given $\alpha \in (0,1)$. As we construct multiple prediction sets, we also provide a control of the FCP (Eq.(5)).

To construct these sets, a natural strategy is to calculate the quantile of order $1-\alpha$ of the scores $s(X_i, R_i^{c+t}, \lbag \mathcal{A}(X_j) \rbag_{j \in \llbracket n+m \rrbracket})$ computed on the calibration set as this is done in the split CP method (see Section 2.1). However, these scores depend on the quantities $\left\{R_i^{c+t}\right\}_{i \in \llbracket n \rrbracket}$ which are unknown. Consequently, they are not computable and we must find another strategy. The following result shows that, if we can bound the ranks of the calibration $\left(R_i^{c+t}\right)_{i \in \llbracket n \rrbracket}$ with high probability, then it is possible to construct marginally valid sets for the test ranks $\left\{R_{n+i}^{c+t}\right\}_{i \in \llbracket m \rrbracket}$.

Theorem 3.5. *Let $\alpha \in (0,1)$, $\delta < \alpha$ and $R_i^-, R_i^+ \in \mathbb{R}$ ($i \in \llbracket n \rrbracket$) be random variables depending on $(X_i, R_i^c)_{i \in \llbracket n \rrbracket}$ and satisfying*

$$\mathbb{P}\big(\forall i \in \llbracket n \rrbracket : R_i^{c+t} \in \left[R_i^-, R_i^+\right]\big) \geqslant 1 - \delta . \tag{6}$$

Let us define the proxy scores for $i \in \llbracket n \rrbracket$:

$$S_i := \max_{r \in \left[R_i^-, R_i^+\right]} s\Big(X_i, r, \lbag X_j \rbag_{j \in \llbracket n+m \rrbracket}\Big) \tag{7}$$

and the associated conformal prediction set:

$$\widehat{\mathcal{C}}_k(x) = \left\{ r : s\big(x, r, \lbag X_i \rbag_{i \in \llbracket n+m \rrbracket}\big) \leqslant S_{(k)} \right\} , \tag{8}$$

$S_{(k)}$ being the k-th smallest proxy score in $S_1, \dots, S_n$.

If Assumption 3.1 is satisfied, then for $k = \lceil (1-\alpha+\delta)(n+1) \rceil$ and any $i \in \llbracket m \rrbracket$:

$$\mathbb{P}\Big(R_{n+i}^{c+t} \in \widehat{\mathcal{C}}_k(X_{n+i})\Big) \geqslant 1 - \alpha .$$

For the score functions $s^{\mathbf{RA}}$ and $s^{\mathbf{VA}}$, the proxy scores of Eq. (7) and the associated conformal sets have explicit expressions which make them easy to compute.

Example 3.6 ($s = s^{\mathbf{RA}}$). The maximum (7) is necessarily attained at the edges of the interval $[R^-, R^+]$:

$$\max_{r \in [R^-, R^+]} s^{\mathbf{RA}}(x, r, \lbag x_i \rbag_i) = \max\big(\left|R^- - \mathcal{A}(x, \lbag x_i \rbag_i)\right|, \left|R^+ - \mathcal{A}(x, \lbag x_i \rbag_i)\right|\big) .$$

The associated conformal set is $\widehat{\mathcal{C}}^{\mathbf{RA}}(x) = \left[\mathcal{A}(x, \lbag X_i \rbag_i) \pm S_{(k)}\right]$.

Example 3.7 ($s = s^{\mathbf{VA}}$). Similarly as the previous example, the maximum is attained at the edges of $[R^-, R^+]$:

$$\max_{r \in [R^-, R^+]} s^{\mathbf{VA}}(x, r, \lbag x_i \rbag_i) = \max_{r \in \{R^-, R^+\}} \Big(\left| \mathcal{A}(x) - \mathtt{R}^{-1}(r, \lbag \mathcal{A}(x_i) \rbag_i) \right| \Big) .$$

The corresponding conformal set is then the interval

$$\widehat{\mathcal{C}}^{\mathbf{VA}}(x) = \Big[\mathtt{R}\Big(\mathcal{A}(x) - S_{(k)}, \Lbag \mathcal{A}(x_i) \Rbag_i\Big), \mathtt{R}\Big(\mathcal{A}(x) + S_{(k)}, \Lbag \mathcal{A}(x_i) \Rbag_i\Big)\Big].$$

We see that contrary to the conformal intervals $\widehat{\mathcal{C}}^{\mathbf{RA}}$ which have the same length for any point x, the length of the conformal interval $\widehat{\mathcal{C}}^{\mathbf{VA}}$ can vary depending on the amount of points around $\mathcal{A}(x)$.

The following result gives a control in high probability of the false coverage rate on the test sample.

Proposition 3.8. *Let $\alpha \in (0,1)$ and consider the prediction sets $\widehat{\mathcal{C}}_k$ defined in Eq.* (8). *Assume Assumption 3.2 holds, let $k = \lceil (1-\alpha)(n+1) \rceil$, then with probability at least $1 - \beta - \delta$:*

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\Big\{R_{n+i}^{c+t} \notin \widehat{\mathcal{C}}_k(X_{n+i})\Big\} \leqslant \alpha + \lambda_{n,m} \ , \tag{9}$$

where $\lambda_{n,m} = \sqrt{\dfrac{\log(C\sqrt{\tau_{n,m}}/\beta)}{\tau_{n,m}}}$, $\tau_{n,m} = \dfrac{nm}{n+m}$ and $C = 4\sqrt{2\pi}$ works.

When the sample sizes increase, then $\lambda_{n,m} \to 0$ and the average error on the test sample is at most α. In practice, using $\alpha' = \alpha - \lambda_{n,m}$ to control the FCP at level α is too conservative. We instead use the numerical procedure explained by Gazin et al. [2024, Remark 2.6] to find a sharper but implicit value for $\lambda_{n,m}$. They use the exact distribution of the FCP to adjust the level of each set accordingly, in order to control it (see Appendix C for details).

3.3 Alternative targets

In the previous section, we focused on the construction of prediction sets for the ranks of test points within the entire data set. In this section, we discuss how to derive from this initial construction, predictions sets for alternative targets. More details are provided in Appendix B.

Calibration points. To obtain predictions sets for the calibration points, we can directly use the high probability bounds (6) on which our method is built. Indeed, by construction the sets $[R_i^-, R_i^+]$ contain R_i^{c+t} and have a null FCP with probability at least $1 - \delta$ for $i \in [\![n]\!]$.

Rank among the test points. A user might also be interested in the ranking of the test points among themselves, rather than within the entire data set. Such sets can be constructed directly from the previous sets (8), by substracting the number of calibration points of smaller rank to each interval boundary (see Corollary B.1).

Top-`K` items. It is also possible to target the top-`K` items by selecting all items whose associated prediction set includes at least one rank smaller than or equal to `K`. This strategy can be sufficient if we target a "top" proportion of the items (see Corollary B.2).

4 Rank of the calibration points

To make our method complete, it remains to find R_i^- and R_i^+ satisfying Eq. (6). Let us recall that $R_i^{c+t} = R_i^c + R_i^t$. Hence, we know that $R_i^{c+t} \in [R_i^c, R_i^c + m]$ and we can use Theorem 3.5 with $R_i^- = R_i^c$ and $R_i^+ = R_i^c + m$. However, using these bounds for instance with the residual conformity function $s^{\mathbf{RA}}$, will increase the length of the conformal set $\widehat{\mathcal{C}}^{\mathbf{RA}}(x)$ by m. In this section, we show that it is possible to find theoretical bounds and practical bounds satisfying Eq. (6) that do not excessively increase the length of the final conformal sets.

4.1 Theoretical bound

As the calibration and test values $(Y_i)_{i \in [\![n+m]\!]}$ are exchangeable (Assumption 3.1), we can expect that a calibration point with a low rank in the calibration set will have a low rank in the test set. The following proposition justifies this intuition.

Proposition 4.1. *Under Assumption 1, for $\delta \in (0,1)$:*

$$\mathbb{P}\Big(R_i^{c+t} \in [R_i^-, R_i^+] \cap [\![n+m]\!], i \in [\![n]\!]\Big) \geqslant 1 - \delta \ ,$$

where $R_i^{\pm} = R_i^c + (m+1)\left(\frac{R_i^c}{n} \pm \sqrt{\frac{\log(C\sqrt{\tau_{n,m}}/\delta)}{\tau_{n,m}}}\right)$, $\tau_{n,m} = \dfrac{nm}{n+m}$, *and* $C = 4\sqrt{2\pi}$ *works.*

The proof can be found in the supplementary and relies on Gazin et al. [2024, Theorem 2.4]. While the naive method provides an interval of length m for the rank R^{c+t}, our approach achieves an interval of length $O(m/\sqrt{\tau_{n,m}})$. The length has been thus reduced by a factor $\sqrt{\tau_{n,m}} \simeq \min(\sqrt{n}, \sqrt{m})$.

This explicit theoretical bound provides qualitative insight into the behavior and evolution of the ranks with (m, n). However, the constant C is not tight (see Figure 1). To obtain finer envelopes, we propose in the following a numerical method to calibrate them, with theoretical guarantee (Proposition 4.2).

4.2 Numerical bounds

The explicit bounds given by Proposition 4.1 are often too conservative in practice. In this section, we construct sharp numerical bounds R_i^-, R_i^+ for the ranks R_i^{c+t} such that the event

$$\left\{\forall\ 1 \leqslant j \leqslant n,\ R_j^{c+t} \text{ is between } R_i^- \text{ and } R_i^+\right\}$$

holds with high probability. Let us denote for $r \in [\![n]\!]$, $R_{(r)}^{c+t}$ the rank in the whole sample of the item of rank r in the calibration. In particular, $R_{(R_i^c)}^{c+t} = R_i^{c+t}$. Our idea is to use that the random vector $R_{srt}^{c+t} = (R_{(1)}^{c+t}, \dots, R_{(n)}^{c+t})$ follows a universal distribution independent of the distribution of the data to simulate them and to estimate an envelope.

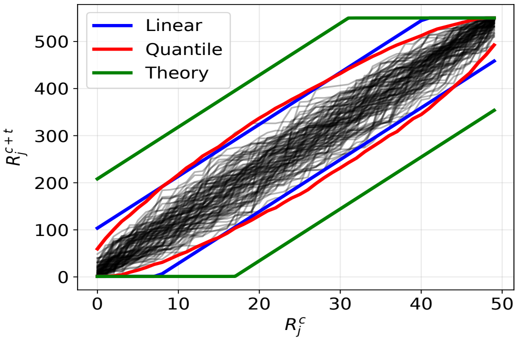


Figure 1: The two envelopes for $n = 50$, $m = 500$, $\delta = 0.1$ and $K = 10^5$. The blue and red lines are respectively the linear and quantile envelopes, and each black curve is an ordered realization of R^{c+t}. The green line is the envelope from Prop. 4.1.

Algorithm 1 Simulation of R_{srt}^{c+t}

```
Draw $n+m$ uniform random variable $U_i$ on $[0,1]$.
for $i = 1, \dots, n$ do
    $\tilde{R}_i^{c+t} \leftarrow \mathrm{R}(U_i, \lbag U_j \rbag_{j \in [\![n+m]\!]})$
end for
Output: Sort $\tilde{R}^{c+t}$
```

This distribution can be drawn using Algorithm 1 thanks to the exchangeability of the data (Assumption 3.1; see Gazin et al., 2024 for more details). An envelope can then be estimated by a Monte-Carlo approach using the empirical cumulative distribution function. We consider two forms of envelope in this work, the linear and the quantile detailed below, which are parameterized by real factors c and γ. To estimate these envelopes, we draw K realizations of vectors following the same distribution as (R_{srt}^{c+t}) by repeating K times Algorithm 1 and estimate the (multidimensional) cumulative function of this sample. Then, we optimize the parameter c or γ for minimizing the width of the envelope (which depends directly on this parameter) while containing a proportion $1-\delta$ of the K trajectories.

Linear envelope: We consider an envelope of the form of Proposition 4.1:

$$\widehat{R}_i^{\pm} = R_i^c + (m+1)\left(\frac{R_i^c}{n} \pm \widehat{c}\right),$$

where $\widehat{c}$ is minimized under the condition that a proportion $(1-\delta)$ of the generated vectors are in the envelope. We then take the minimum with $n+m$ and the maximum with 1 to keep the bounds in $[\![n+m]\!]$.

Quantile envelope: For an item of rank $r = R_i^c$, we propose to choose the bound $\widehat{R}_i^+$ (respectively $\widehat{R}_i^-$) as the empirical quantile of order $1-\widehat{\gamma}$ (resp. $\widehat{\gamma}$) of the simulated ranks $R_{(r)}^{c+t}$, i.e. the ranks in the whole sample of the points of rank r in the calibration. As we want uniform control of these bounds, we need to adjust γ. Hence, the parameter γ is maximized under the condition that a proportion $1-\delta$ of the generated vectors are in the envelope. Algorithm 2 in Appendix C gives the full process to construct this envelope.

As it can be observed in Figure 1, the quantile envelope is smaller for the low and high rank points while the linear envelope is smaller for the points of medium rank. Furthermore, they both outperform the theoretical envelope.

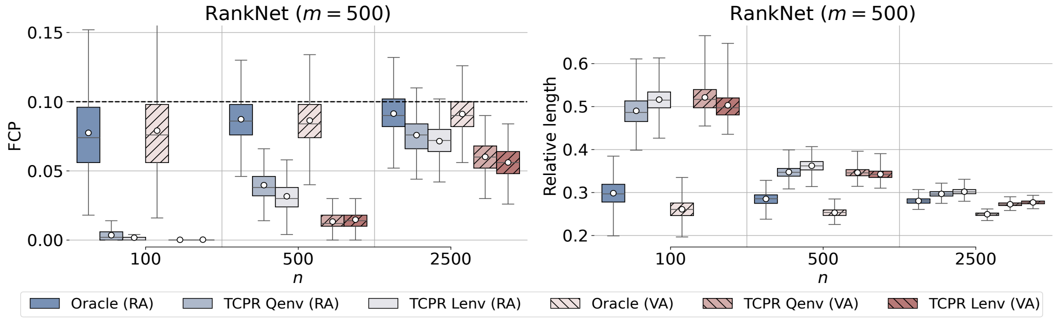


Figure 2: Synthetic data: FCP and relative lengths obtained for RankNet with the (**RA**) and (**VA**) score, for the quantile (*Qenv*) and linear (*Lenv*) envelopes when $m = 500$ and $n \in \{100, 500, 2500\}$. White circles represent the means.

Monte-Carlo guarantees: The estimation of the envelopes brings another source of error in our procedure. In the following proposition, we show that this error can be easily controlled and the parameter K can be chosen large enough to make it negligible.

Proposition 4.2. *Let $K, n, m \in \mathbb{N}^*$ and $(R^{(k)})_{k\in[\![K]\!]}$ a K-sample of ordered ranks drawn from Algorithm 1 with parameters n and m. If $(\widehat{R}_i^{\pm})_{i\in[\![n]\!]}$ is an estimated envelope, possibly depending on $(R^{(k)})_k$ such that, almost surely $\frac{1}{K}\sum_{k=1}^{K}\mathbf{1}\left\{\exists\, i : R_i^{(k)} \notin \left[\widehat{R}_i^-, \widehat{R}_i^+\right]\right\} \leqslant \delta$, then:*

$$\mathbb{P}\left[\forall i : R_i^{c+t} \in \left[\widehat{R}^-_{R_i^c}, \widehat{R}^+_{R_i^c}\right]\right] \geqslant 1 - \delta - 4\sqrt{\frac{\log nK}{K}}\,. \tag{10}$$

The proof is given in Appendix A.5. Notice that this result is not specific to the linear or the quantile envelope we consider but rather to the Monte-Carlo procedure itself. It is indeed based on a control of the multi-dimensional empirical cumulative distribution function from Naaman [2021].

5 Experiments

In this section, we empirically evaluate the performance of our CP method referred to as Transductive CP for Ranking (TCPR) on different datasets and algorithms. We construct prediction sets using the score function $s^{\mathbf{RA}}$ or $s^{\mathbf{VA}}$ and with the linear or the quantile envelopes described in Section 4.2. We compare ourselves to the *Oracle* method, which is the conformal method using non-conformity scores calculated with the unobserved true ranks $\left\{R_i^{c+t}\right\}_{i\in[\![n]\!]}$. This comparison allows for an evaluation of the effect of the envelope.

The code of our method is available at `https://github.com/pierreHmbt/transductive-conformal-inference-for-ranking`.

Ranking algorithms: We use in the experiments the following algorithms: RankNet [Burges et al., 2005], LambdaMART [Wu et al., 2010] and RankSVM [Herbrich et al., 1999]. These methods learn a score function of the features from pairwise comparisons. In the Appendix, we also use the Balanced Rank Estimation (BRE) [Wauthier et al., 2013] which ranks items using a limited number of comparisons.

Parameters: The parameters α, $1-\beta$ and δ, equal to respectively, the probability of miscoverage, the probability to control the FCP at level α, and the probability of the quantile envelope, are set to, respectively, 0.1, 0.75 and 0.02, in all our experiments.

Metrics: We use the following metrics to evaluate our method. **1)** The FCP of the new ranks (Eq (5)). **2)** The *relative length* of the prediction sets defined as the average size of the sets divided by $n + m$. **3)** The *oracle ratio* defined as the ratio between the size of the set from TCPR and from the oracle.

Some metrics are not presented in this section and can be found in Appendix D with additional information on the data sets and the algorithms, such as their hyperparameters and architecture.

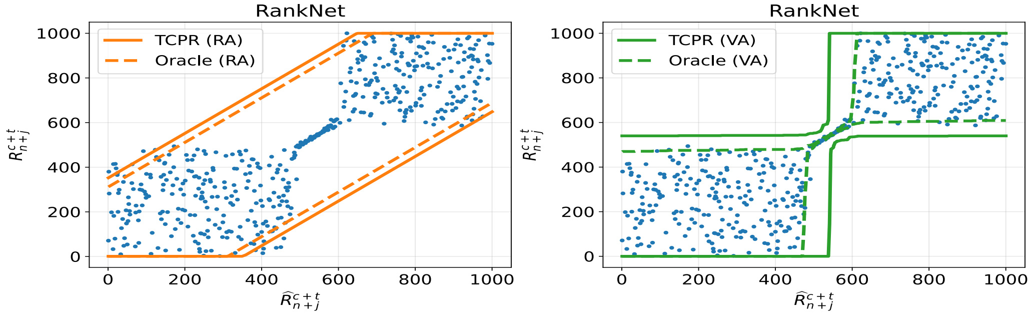


Figure 3: Data from Section 5.1: True ranks R^{c+t}_{n+j} in function of their predicted rank $\widehat{R}^{c+t}_{n+j}$ by RankNet and their prediction sets with scores $s^{\mathbf{RA}}$ and $s^{\mathbf{VA}}$ for $n = m = 500$.

5.1 Synthetic data

Data generation: We use the same generation process as Pedregosa et al. [2012] with slight modifications to match our setting. In detail, we draw $n + m$ pair of points (X_i, Y_i) where X_i is a Gaussian vector $\mathcal{N}(0, I_d)$ of dimension $d = 5$, $Y_i = \big(1 + \exp(-w^T X_i)\big)^{-1} + \varepsilon_i$ with $w \sim \mathcal{N}(0, I_d)$ and $\varepsilon_i \sim \mathcal{N}(0, 0.07)$. The rank R_i of an observation is defined as the rank of the corresponding Y_i. The sizes of the calibration and test sets are chosen among $n, m \in \{100, 500, 2500\}$ and for each pair (n, m), we compute the FCP and the relative length on $B = 1000$ generated data sets.

Results: Figure 2 displays the FCP and the relative length for RankNet with the score function $s^{\mathbf{RA}}$ or $s^{\mathbf{VA}}$ and the two proposed envelopes when $m = 500$. Results for other values of n and m and for LambdaMART, RankSVM and BRE are in Appendix D.

We first remark that the FCPs are correctly controlled by our method as they remain below $\alpha = 0.1$ at least $100 \cdot (1 - \beta)\% = 75\%$ of the time, as indicated by the upper boundary of the boxes. It is important to note, however, that due to the necessity of using proxy scores, our method tends to be conservative for small values of n. For example, when $n = 100$, we see that the FCPs obtained with TCPR are close to 0 whereas those obtained with the oracle method are in average near 0.07. Nevertheless, this conservatism diminishes as the calibration size n increases with FCPs close to 0.1 when $n = 2500$. In terms of relative length, the prediction sets constructed using the quantile envelope are, on average, consistently smaller than those obtained with the linear envelope for FCPs closer to 0.1. The quantile envelope is therefore preferable. Finally, notice that, at least for RankNet, using $s^{\mathbf{RA}}$ or $s^{\mathbf{VA}}$ gives sets with similar sizes.

Adaptivity of score $s^{\mathbf{VA}}$. To further highlight the adaptability of score $s^{\mathbf{VA}}$ relatively to $s^{\mathbf{RA}}$, we consider another data set of size $n = m = 500$. Each pair (X_i, Y_i) is defined by $Y_i = X_i + \varepsilon_i$ where $X_i \sim$ Beta$(.04, .04)$ and $\varepsilon_i \sim \mathcal{N}(0, .07)$. Here, because the random variables $(X_i)_i$ follow a Beta distribution with parameters lower than 1, the probability of observing a value near 0 or 1 is much larger than the probability of observing a value near 0.5. As there is more observations close to these values, RankNet ranks with more difficultly the observations near 0 or 1 than the others. However, as the prediction sets constructed with $s^{\mathbf{RA}}$ have all the same size, they do not reflect this difficulty (see an example in Figure 3 left panel). On the contrary, the sets returned with $s^{\mathbf{VA}}$ are narrower when the ranking is easier (close to $500 = 0.5 \cdot (n + m)$) and wider for small and large ranks (Figure 3 right panel).

5.2 Real data: Yummly-`10k`

Dataset: We evaluate our approach on the Yummly `Food-10k` data set which consists in 12624 images of dishes embedded in $\mathbb{R}^{101}$. These embeddings have been constructed to reflect similarities in taste among the dishes (see Wilber et al., 2015[1] for a complete description).

Ranking task: As in Canal et al. [2019], to define a full order of these items, we select an (unknown) preference point x^* and define this ordering using the distance to this point. The $(Y_i)_i$ are thus these distances. We use RankNet, LambdaMART, and RankSVM to rank the dishes and our method

[1] Companion website: `http://vision.cornell.edu/se3/projects/concept-embeddings`

	Rank error	FCP (**RA**)	RL (**RA**)	OR (**RA**)	FCP (**VA**)	RL (**VA**)	OR (**VA**)
RankNet	675.9	0.069 ± 0.005	0.306 ± 0.003	1.095 ± 0.003	0.069 ± 0.005	0.312 ± 0.003	1.104 ± 0.005
LambdaMART	714.8	0.072 ± 0.004	0.324 ± 0.003	1.087 ± 0.005	0.071 ± 0.005	0.323 ± 0.003	1.089 ± 0.004
RankSVM	751.3	0.076 ± 0.005	0.400 ± 0.004	1.064 ± 0.003	0.071 ± 0.004	0.408 ± 0.004	1.079 ± 0.005

Table 1: Results on the Yummly-10k data set. RL is for Relative Length and OR for Oracle Ratio.

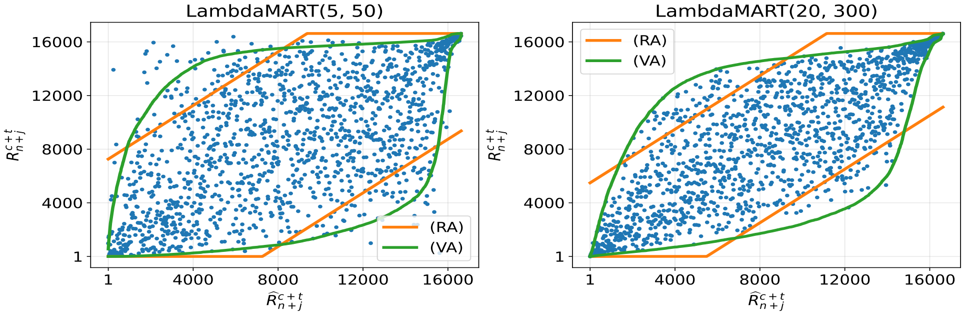


Figure 4: Ranks predicted by LambdaMART(leaves, trees) in function of the ranks predicted by a larger model of 400 trees of 20 leaves.

TCPR with the quantile envelope to construct the prediction sets. We divide the data into a training, calibration and test sets of respective size $n_{tr} = 2624$, $n = 2000$ and $m = 8000$. We repeat this splitting 10 times to compute the metrics.

Results: Table 1 reports the values of the different metrics. Firstly, all methods are well-calibrated with FCP consistently remaining below the chosen threshold 0.1. Secondly, the size of the intervals decreases as performance improves, as reflected by the rank error and the relative length metrics. The impact of the envelope is reflected in the ratio with the oracle. This impact is negligible as the ratio remains close to 1. We can however notice that it increases as model performance improves. Indeed, the more effective the algorithm, the more the error introduced by the envelope becomes noticeable and impacts the size of the prediction set. Additional results are given in Appendix E.1.

5.3 Real data: Anime recommendation LTR data set

Dataset and ranking task: This dataset is composed of features of 16681 movies, characteristics of 15163 users and 10^6 ratings associated with a tuple (user, movie) with values ranging from 0 to 10. Given the characteristics of a new user, the objective is to produce an ordered list of all the movies ranked by the user's level of interest. We aim to quantify the uncertainty of a smaller model relative to the performance of a larger one, which serves as a reference and defines the ground-truth full ordering of the items, i.e., R_i^{c+t}. More details are provided in Appendix E.2.

Results: As shown in Figure 4, the size of the prediction sets for both $(\mathbf{RA})$ and $(\mathbf{VA})$ decreases as the model quality improves. In particular, LambdaMART (leaves=20, trees=300) yields results that are closer to those of the larger model (LambdaMART with 20 leaves and 400 trees) than the smaller ranker (LambdaMART with 5 leaves and 50 trees). This approach has the advantage of allowing for an easy comparison between the two methods without requiring the large model to be run on all movies which can be expensive. We can also point out again the better adaptivity of the score $s^{\mathbf{VA}}$ The metrics and comparisons with other architectural configurations are presented in Appendix E.2.

6 Conclusion

We have developed a conformal prediction method that quantifies the error of an algorithm ranking m new items among n previously ranked items. Our approach is based on constructing an envelope around these n ranks to quantify the algorithm's error. Both theoretical and empirical evidence demonstrate that this envelope does not significantly impact the size of the intervals, especially when n is large relative to m. One limitation of our work is that we only focus on full ranking. It would therefore be interesting to extend our method to a partial ranking framework. Another important line of research would be to cover top-k algorithms. Finally, the dependence of our method on the choice of the envelope we select should be investigated. We expect that a choice more adapted to the problem considered could improve the size of the prediction sets.

Acknowledgements

The authors acknowledge Ulysse Gazin and Etienne Roquain for helpful comments. G.B. gratefully acknowledges funding from the grants ANR-21-CE23-0035 (ASCAI), ANR-19-CHIA-0021-01 (BISCOTTE) and ANR-23-CE40-0018-01 (BACKUP) of the French National Research Agency ANR. P.H. gratefully acknowledges the Emergence project MARS of Sorbonne Université.

References

Yunbo Cao, Jun Xu, Tie-Yan Liu, Hang Li, Yalou Huang, and Hsiao-Wuen Hon. Adapting ranking svm to document retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193, 2006.

Nathan N Liu and Qiang Yang. Eigenrank: a ranking-oriented approach to collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 83–90, 2008.

Yang Liu, Jian-Wu Bi, and Zhi-Ping Fan. Ranking products through online reviews: A method based on sentiment analysis technique and intuitionistic fuzzy set theory. *Information Fusion*, 36: 149–161, 2017.

Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, and Klaus Brinker. Multilabel classification via calibrated label ranking. *Machine learning*, 73:133–153, 2008.

Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

Anastasios N Angelopoulos, Karl Krauth, Stephen Bates, Yixin Wang, and Michael I Jordan. Recommendation systems with distribution-free reliability guarantees. In *Conformal and Probabilistic Prediction with Applications*, pages 175–193. PMLR, 2023a.

Yunpeng Xu, Wenge Guo, and Zhi Wei. Conformal ranked retrieval. *arXiv preprint arXiv:2404.17769*, 2024.

Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill™: a bayesian skill rating system. *Advances in neural information processing systems*, 19, 2006.

Josh Alman and Dylan McKay. Theoretical foundations of team matchmaking. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 1073–1081, 2017.

Tom Minka, Ryan Cleven, and Yordan Zaykov. Trueskill 2: An improved bayesian skill rating system. *Technical Report*, 2018.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *Machine Learning: ECML 2002: 13th European Conference on Machine Learning Helsinki, Finland, August 19–23, 2002 Proceedings 13*, pages 345–356. Springer, 2002.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. *Advances in neural information processing systems*, 32, 2019.

Anastasios N Angelopoulos, Stephen Bates, et al. Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4):494–591, 2023b.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523): 1094–1111, 2018.

Matteo Fontana, Gianluca Zeni, and Simone Vantini. Conformal prediction: a unified review of theory and new challenges. *Bernoulli*, 29(1):1–23, 2023.

Valery Manokhin. Awesome conformal prediction. `https://github.com/valeman/awesome-conformal-prediction`, 2024. Github repository.

Vladimir Vovk. Transductive conformal predictors. In *Artificial Intelligence Applications and Innovations: 9th IFIP WG 12.5 International Conference, AIAI 2013, Paphos, Cyprus, September 30–October 2, 2013, Proceedings 9*, pages 348–360. Springer, 2013.

Paulo C. Marques F. Universal distribution of the empirical coverage in split conformal prediction. *arXiv preprint*, 2024.

Kexin Huang, Ying Jin, Emmanuel Candes, and Jure Leskovec. Uncertainty quantification over graph with conformalized graph neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

Ulysse Gazin, Gilles Blanchard, and Etienne Roquain. Transductive conformal inference with adaptive scores. In *International Conference on Artificial Intelligence and Statistics*, pages 1504–1512. PMLR, 2024.

Ulysse Gazin. Asymptotics for conformal inference. *arXiv preprint arXiv:2409.12019*, 2024.

R. Herbrich, T. Graepel, and K. Obermayer. Support vector learning for ordinal regression. In *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, volume 1, pages 97–102, 1999. doi: 10.1049/cp:19991091.

Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

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.

Christopher Burges, Robert Ragno, and Quoc Le. Learning to rank with nonsmooth cost functions. *Advances in neural information processing systems*, 19, 2006.

Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. Adapting boosting for information retrieval measures. *Information Retrieval*, 13:254–270, 2010.

Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11 (23-581):81, 2010.

Zhen Qin, Le Yan, Honglei Zhuang, Yi Tay, Rama Kumar Pasumarthi, Xuanhui Wang, Michael Bendersky, and Marc Najork. Are neural rankers still outperformed by gradient boosted decision trees? In *International conference on learning representations*, 2021.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136, 2007.

Hang Li. A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems*, 94(10):1854–1862, 2011.

Fabian Wauthier, Michael Jordan, and Nebojsa Jojic. Efficient ranking from pairwise comparisons. In *International Conference on Machine Learning*, pages 109–117. PMLR, 2013.

Nir Ailon. An active learning algorithm for ranking from pairwise preferences with an almost optimal query complexity. *Journal of Machine Learning Research*, 13(1), 2012.

Reinhard Heckel, Nihar B. Shah, Kannan Ramchandran, and Martin J. Wainwright. Active ranking from pairwise comparisons and when parametric assumptions do not help. *The Annals of Statistics*, 47(6):3099–3126, 2019.

Sahand Negahban, Sewoong Oh, and Devavrat Shah. Iterative ranking from pair-wise comparisons. *Advances in neural information processing systems*, 25, 2012.

Uriel Feige, Prabhakar Raghavan, David Peleg, and Eli Upfal. Computing with noisy information. *SIAM Journal on Computing*, 23(5):1001–1018, 1994.

Mark Braverman and Elchanan Mossel. Noisy sorting without resampling. In *ACM-SIAM Symposium on Discrete Algorithms*, 2007.

Mark Braverman and Elchanan Mossel. Sorting from noisy information. *CoRR*, abs/0910.1191, 2009.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

R Duncan Luce et al. *Individual choice behavior*, volume 4. Wiley New York, 1959.

Yue Liu, Ethan X Fang, and Junwei Lu. Lagrangian inference for ranking problems. *Operations research*, 71(1):202–223, 2023.

Chao Gao, Yandi Shen, and Anderson Y Zhang. Uncertainty quantification in the bradley–terry–luce model. *Information and Inference: A Journal of the IMA*, 12(2):1073–1140, 2023.

Jianqing Fan, Zhipeng Lou, Weichen Wang, and Mengxin Yu. Ranking inferences based on the top choice of multiway comparisons. *Journal of the American Statistical Association*, 120(549): 237–250, 2025.

Michael Naaman. On the tight constant in the multivariate dvoretzky–kiefer–wolfowitz inequality. *Statistics & Probability Letters*, 173:109088, 2021.

Fabian Pedregosa, Elodie Cauvet, Gaël Varoquaux, Christophe Pallier, Bertrand Thirion, and Alexandre Gramfort. Learning to rank from medical imaging data. In *Machine Learning in Medical Imaging: Third International Workshop, MLMI 2012, Held in Conjunction with MICCAI 2012, Nice, France, October 1, 2012, Revised Selected Papers 3*, pages 234–241. Springer, 2012.

Michael Wilber, Iljung S Kwak, David Kriegman, and Serge Belongie. Learning concept embeddings with combined human-machine expertise. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 981–989, 2015.

Gregory Canal, Andy Massimino, Mark Davenport, and Christopher Rozell. Active embedding search via noisy paired comparisons. In *International Conference on Machine Learning*, pages 902–911. PMLR, 2019.

Aryeh Dvoretzky, Jack Kiefer, and Jacob Wolfowitz. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, pages 642–669, 1956.

A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

Appendix

A Proofs

A.1 Proof of Theorem 3.5

Let us denote $E_\delta = \big\{\forall i \in [\![n]\!] : R_i^{c+t} \in \big[R_i^-, R_i^+\big]\big\}$. On the event E_δ, for all $i \in [\![n]\!]$, we have:

$$S_i^{\texttt{true}} := s\big(X_i, R_i^{c+t}, (X_k)_{k\in[\![n+m]\!]}\big) \leqslant \max_{r\in[R_i^-, R_i^+]} s\big(X_i, r, (X_k)_{k\in[\![n+m]\!]}\big) := S_i\,.$$

Hence, the quantiles of the "true" scores $S_i^{\texttt{true}}$ are upper bounded by the quantiles of the proxy scores $\{S_i\}_{i\in[\![n]\!]}$, i.e., for all $k \in [\![n]\!]$:

$$S_{(k)}^{\texttt{true}} \leqslant S_{(k)}\,. \tag{11}$$

Then, for $i \in [\![m]\!]$:

$$\begin{aligned}\mathbb{P}\Big(R_{n+i}^{c+t} \notin \widehat{\mathcal{C}}(X_{n+i})\Big) &\leqslant \mathbb{P}\Big(R_{n+i}^{c+t} \notin \widehat{\mathcal{C}}(X_{n+i}) \cap E_\delta\Big) + \mathbb{P}(E_\delta^c)\\ &\leqslant \mathbb{P}\big(s\big(X_{n+i}, R_{n+i}^{c+t}, (X_j)_{j\in[\![n+m]\!]}\big) > S_{(k)} \cap E_\delta\big) + \mathbb{P}(E_\delta^c)\\ &\leqslant \mathbb{P}\Big(s\big(X_{n+i}, R_{n+i}^{c+t}, (X_j)_{j\in[\![n+m]\!]}\big) > S_{(k)}^{\texttt{true}}\Big) + \delta \leqslant \alpha - \delta + \delta = \alpha\,.\end{aligned}$$

To pass from the second to the third line, we have used (11) which is satisfied on the event E_δ. The last inequality is obtained using that if the score function $s(x, r, (x_j)_j)$ is invariant by permutation of its last argument and the vectors $(X_i, R_i^{c+t})_{i\in[\![n+m]\!]}$ are exchangeable, then the true scores $\{S_i^{\texttt{true}}\}_{i\in[\![n]\!]}$ are exchangeable. Therefore, we can use Theorem 1 with $k = \lceil (n+1)(1-\alpha+\delta)\rceil$.

A.2 Restatement of Gazin et al. [2024, Theorem 2.4]

We restate here Theorem 2.4 of Gazin et al. [2024] which controls the empirical cumulative distribution function of conformal p-values.

Theorem A.1. *Let $n, m \geqslant 2$, $V_1, \ldots, V_{n+m}$ be exchangeable real random variables having no ties almost surely. Let us, for $1 \leqslant i \leqslant n$, define the conformal p-values:*

$$p_i = \frac{1}{m+1}\left(1 + \sum_{j=1}^{m} \mathbf{1}\{V_i \geqslant V_{n+j}\}\right) \tag{12}$$

and $\widehat{F}_n(t) := n^{-1}\sum_{i=1}^{n} \mathbf{1}\{p_i \leqslant t\}$ be their empirical cumulative distribution function. Then, for $\beta \in (0,1)$:

$$\mathbb{P}\left[\sup_{t\in[0,1]}\Big|\widehat{F}_n(t) - I_m(t)\Big| > \sqrt{\frac{\log(1/\beta) + \log\big(1 + \sqrt{2\pi\tau_{n,m}}\big)}{2\tau_{n,m}}}\right] \leqslant 2\beta\,, \tag{13}$$

where $\tau_{n,m} = \frac{nm}{n+m}$ and $I_m(t) = \lfloor (m+1)t\rfloor/(m+1)$.

Proof. The concentration inequality is obtained from Gazin et al. [2024, Theorem 2.4]. The lower deviation was enounced in its proof (Equation (38)). We apply the theorem with $r = 1$ and upper bound $\tau_{n,m}/\sqrt{n+m}$ by $\sqrt{\tau_{n,m}}/2$:

$$\begin{aligned}\Psi(1) = 1 \wedge \left(\frac{\log(1/\beta) + \log(1 + \sqrt{2\pi}\frac{2\tau_{n,m}}{\sqrt{n+m}})}{2\tau_{n,m}}\right) &\leqslant \frac{\log(1/\beta) + \log\big(1 + \sqrt{2\pi\tau_{n,m}}\big)}{2\tau_{n,m}}\\ &\leqslant \frac{\log(2\sqrt{2\pi\tau_{n,m}}/\beta)}{2\tau_{n,m}}\,.\end{aligned}$$

The first term is the deviation obtained by Gazin et al. [2024]. The first upper bound is the one appearing in equation (13). The second upper bound is used to get Proposition 3.8 □

A.3 Proof of Proposition 3.8

On the event $\big\{\forall i \in [\![n]\!] : R_i^{c+t} \in \big[R_i^-, R_i^+\big]\big\}$, we have:

$$S_i^{\texttt{true}} := s\big(X_i, R_i^{c+t}, (X_k)_{k\in[\![n+m]\!]}\big) \leqslant \max_{r\in[R_i^-,R_i^+]} s\big(X_i, r, (X_k)_{k\in[\![n+m]\!]}\big) := S_i \,.$$

Hence, as earlier the quantiles of the "true" scores $S_i^{\texttt{true}}$ are upper bounded by the quantiles of the proxy scores $\{S_i\}_{i\in[\![n]\!]}$, i.e., for all $k \in [\![n]\!]$:

$$S_{(k)}^{\texttt{true}} \leqslant S_{(k)} \,. \tag{14}$$

Then, on this event:

$$\sum_{i=1}^{m} \mathbf{1}\Big\{R_{n+i}^{c+t} \notin \widehat{\mathcal{C}}_k(X_{n+i})\Big\} = \sum_{i=1}^{m} \mathbf{1}\big\{S_{n+i}^{\texttt{true}} > S_{(k)}\big\} \leqslant \sum_{i=1}^{m} \mathbf{1}\Big\{S_{n+i}^{\texttt{true}} > S_{(k)}^{\texttt{true}}\Big\} \,. \tag{15}$$

If the true scores have no ties, we can directly apply Theorem A.1 to obtain a control of the FCP. The following technical details allow to control it in the case where the score function is discrete, for instance when $s = s^{\mathbf{RA}}$. Let us introduce

$$\Gamma := \min_{\substack{1\leqslant i,j\leqslant n+m \\ S_i^{\texttt{true}}\neq S_j^{\texttt{true}}}} \big|S_i^{\texttt{true}} - S_j^{\texttt{true}}\big|$$

and an independent family of random variables γ_i for $i \in [\![n+m]\!]$ (also independent of $(S_i^{\texttt{true}})_{i\in[\![n+m]\!]}$) following a uniform distribution on $(-\Gamma/2, \Gamma/2)$. Define $\widetilde{S}_i := S_i^{\texttt{true}} + \gamma_i$ for $i \in [\![n+m]\!]$. Observe that Γ depends on the true scores, but is invariant by permutation of these scores. Hence, the "perturbed" scores $\widetilde{S}_i$ are exchangeable and have (almost surely) no ties [2]. Moreover for $i \in [\![m]\!]$:

$$\begin{aligned}
\Big\{S_{n+i}^{\texttt{true}} > S_{(k)}^{\texttt{true}}\Big\} &= \left\{\sum_{j=1}^{n} \mathbf{1}\big\{S_{n+i}^{\texttt{true}} > S_j^{\texttt{true}}\big\} \geqslant k\right\} \\
&\subseteq \left\{\sum_{j=1}^{n} \mathbf{1}\big\{S_{n+i}^{\texttt{true}} + \gamma_{n+i} > S_j^{\texttt{true}} + \gamma_j\big\} \geqslant k\right\} \\
&= \left\{\sum_{j=1}^{n} \mathbf{1}\Big\{\widetilde{S}_{n+i} > \widetilde{S}_j\Big\} \geqslant k\right\} = \Big\{\widetilde{S}_{n+i} > \widetilde{S}_{(k)}\Big\} \,.
\end{aligned}$$

The main argument of these inequalities is that by construction of the random variables γ_i, for all i, j, the event $\big\{S_i^{\texttt{true}} > S_j^{\texttt{true}}\big\}$ is included in $\Big\{\widetilde{S}_i > \widetilde{S}_j\Big\}$.

Then we can upper bound the FCP using (15):

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\Big\{R_{n+i}^{c+t} \notin \widehat{\mathcal{C}}_k(X_{n+i})\Big\} \leqslant \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\Big\{\widetilde{S}_{n+i} > \widetilde{S}_{(k)}\Big\} \,. \tag{16}$$

We want to apply Theorem A.1 to the exchangeable random variables $\widetilde{S}$, which have no ties. Let us express the FCP in term of conformal p-values. We introduce for $i \in [\![m]\!]$

$$\widetilde{p}_i := \frac{1}{n+1}\left(1 + \sum_{j=1}^{n} \mathbf{1}\Big\{\widetilde{S}_{n+i} \geqslant \widetilde{S}_j\Big\}\right)$$

and for $t \in (0,1)$

$$\widetilde{F}_m(t) := \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\{\widetilde{p}_i \leqslant t\} \,.$$

[2] Γ is not defined if all scores are equal. However, in this case the FCP is null and then upper bounded by α.

Then for $i \in [\![m]\!]$, the following events are equal:

$$\Big\{\widetilde{S}_{n+i} > \widetilde{S}_{(k)}\Big\} = \Big\{\sum_{j=1}^{n} \mathbf{1}\Big\{\widetilde{S}_{n+i} > \widetilde{S}_j\Big\} \geqslant k\Big\} = \Big\{\sum_{j=1}^{n} \mathbf{1}\Big\{\widetilde{S}_{n+i} \geqslant \widetilde{S}_j\Big\} \geqslant k\Big\}$$
$$= \Big\{\sum_{j=1}^{n} \mathbf{1}\Big\{\widetilde{S}_{n+i} \geqslant \widetilde{S}_j\Big\} > k-1\Big\} = \Big\{\widetilde{p}_i > k/(n+1)\Big\} ,$$

where we have used that the scores $\widetilde{S}$ have no ties. Injecting these equalities into (16), we get:

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\Big\{R^{c+t}_{n+i} \notin \widehat{\mathcal{C}}_k(X_{n+i})\Big\} \leqslant 1 - \widetilde{F}_m\Big(k/(n+1)\Big) .$$

Then, with probability $1-(\beta+\delta)$, using (13) (inverting n and m), we get:

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\Big\{R^{c+t}_{n+i} \notin \widehat{\mathcal{C}}_k(X_{n+i})\Big\} \leqslant 1 - I_n\Big(k/(n+1)\Big) + \lambda_{n,m} = 1 - \frac{k}{n+1} + \lambda_{n,m} \leqslant \alpha + \lambda_{n,m} ,$$

where $\lambda_{n,m} = \sqrt{\dfrac{\log(C\sqrt{\tau_{n,m}}/\beta)}{\tau_{n,m}}}$. □

A.4 Proof of Proposition 4.1

Let us apply Theorem A.1 to $V_i = Y_i$ for $1 \leqslant i \leqslant n+m$. Then the p-value p_i-s are affine transformations of the ranks of the calibration point in the test sample:

$$p_i = \frac{1}{m+1}\big(1 + R^t_i\big) ,$$

where we recall that $R^t_i := \mathtt{R}\Big(Y_i, \{Y_{n+j}\}_{j\in[\![m]\!]}\Big)$. Let us then rewrite the empirical cumulative distribution function $\widehat{F}_n$ in function of R^t_i, for $t \in (0,1)$:

$$\widehat{F}_n(t) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\big\{R^t_i \leqslant (m+1)t - 1\big\} .$$

Let us now remark that for some $u \in \mathbb{R}_+$, it is equivalent for a calibration sample point j to have its rank in the test set smaller than u than to have more points than its rank in the calibration with a rank smaller than u:

$$\forall j \in [\![n]\!] : \qquad R^t_j \leqslant u \Longleftrightarrow \sum_{i=1}^{n} \mathbf{1}\big\{R^t_i \leqslant u\big\} \geqslant R^c_j \Longleftrightarrow n\widehat{F}_n\big((u+1)/(m+1)\big) \geqslant R^c_j .$$

We point out that the first equivalence is satisfied as the order of the items in the calibration set is conserved when considering their insertion ranks in the test set: for all $1 \leqslant i, j \leqslant n$, $R^c_i \leqslant R^c_j \Rightarrow R^t_i \leqslant R^t_j$. Thus, for $j \in [\![n]\!]$, if $R^t_j \leqslant u$, the R^c_j calibration items $i \in [\![n]\!]$ satisfying $R^c_i \leqslant R^c_j$ also satisfy $R^t_i \leqslant u$, showing the direct implication ($\Rightarrow$) in the last display. Conversely, if k is the number of calibration items $i \in [\![n]\!]$ satisfying $R^t_i \leqslant u$, they are necessarily exactly the items corresponding to the k smallest calibration ranks. Otherwise, there would be a calibration item $j \in [\![n]\!]$ with $R^c_j = k' > k$ and $R^t_j \leqslant u$, implying (by the previous token) that $\sum_{i=1}^{n} \mathbf{1}\{R^t_i \leqslant u\} \geqslant k' > k$, a contradiction. Thus any calibration item with $R^c_j \leqslant k$ satisfies $R^t_j \leqslant u$.

Thus, with probability $1-2\delta$, thanks to (13), for all $t \in (0,1)$:

$$\Big|\widehat{F}_n(t) - I_m(t)\Big| \leqslant \lambda, \quad \text{where} \quad \lambda = \sqrt{\frac{\log(1/\delta) + \log\big(1 + \sqrt{2\pi\tau_{n,m}}\big)}{2\tau_{n,m}}} .$$

Assume this event is satisfied. Then in particular, for $u = (m+1)\big(R^c_j/n + \lambda\big)$:

$$\begin{aligned} n\widehat{F}_n\big((u+1)/(m+1)\big) &= n\widehat{F}_n\big(R^c_j/n + \lambda + 1/(m+1)\big) \\ &\geqslant nI_m\big(R^c_j/n + \lambda + 1/(m+1)\big) - n\lambda \\ &= n\big\lfloor (m+1)\big(R^c_j/n + \lambda\big) + 1\big\rfloor/(m+1) - n\lambda \\ &\geqslant n\big(R^c_j/n + \lambda\big) - n\lambda \geqslant R^c_j . \end{aligned}$$

Thus, $R_j^t \leqslant (m+1)\big(R_j^c/n+\lambda\big)$ for $1 \leqslant j \leqslant n$. Let us now lower bound these ranks. Similarly, we have the equivalence for $u \in \mathbb{R}$ and $1 \leqslant j \leqslant n$:

$$R_j^t > u \iff \sum_{i=1}^{n} \mathbf{1}\big\{R_i^t \leqslant u\big\} \leqslant R_j^c \iff n\widehat{F}_n\big((u+1)/(m+1)\big) \leqslant R_j^c \,.$$

With $u = (m+1)\big(R_j^c/n-\lambda\big)$, we have $n\widehat{F}_n\big((u+1)/(m+1)\big) \leqslant R_j^c$ and then $R_j^t \geqslant (m+1)\big(R_j^c/n-\lambda\big)$ for $1 \leqslant j \leqslant n$ which concludes the proof. □

A.5 Proof of Proposition 4.2

The proof is based on the concentration bound of Naaman [2021] recalled below which is a recent multi-dimensional version of the Dvoretzky–Kiefer–Wolfowitz (DKW) inequality [Dvoretzky et al., 1956].

Theorem A.2 (Naaman, 2021)**.** *Let $X^{(1)}, \dots X^{(K)}$ i.i.d. random vectors in $\mathbb{R}^n$. Then for $t \geqslant 0$,*

$$\mathbb{P}\left[\sup_{\theta\in\mathbb{R}^n}\Big|F(\theta)-\widehat{F}_K(\theta)\Big| \geqslant t\right] \leqslant n(K+1)e^{-2Kt^2}\,, \tag{17}$$

where $F(\theta) = \mathbb{P}\Big[X_i^{(1)} \leqslant \theta_i, \forall i\Big]$ and $\widehat{F}_K$ is the empirical cdf defined by:

$$\widehat{F}_K(\theta) = \frac{1}{K}\sum_{k=1}^{K}\mathbf{1}\Big\{X_i^{(k)} \leqslant \theta_i, \forall i\Big\}\,.$$

Proof of Proposition 4.2. For $\theta \in \mathbb{R}^n$, let us denote

$$F(\theta) = \mathbb{P}\Big[\forall i \in [\![n]\!] : R_{(i)}^{c+t} \leqslant \theta_i\Big]$$

and $\widehat{F}_K$, the empirical cdf:

$$\widehat{F}_K(\theta) = \frac{1}{K}\sum_{k=1}^{K}\mathbf{1}\Big\{R_i^{(k)} \leqslant \theta_i, \forall i \in [\![n]\!]\Big\}\,.$$

Let us now remark that:

$$\begin{aligned}\mathbb{P}\Big[\forall i \in [\![n]\!] : R_i^{c+t} \in \Big[R^-_{R_i^c}, R^+_{R_i^c}\Big]\Big] &= \mathbb{P}\Big[\forall i \in [\![n]\!] : R_{(i)}^{c+t} \in \big[R_i^-, R_i^+\big]\Big] \\ &= \mathbb{E}\big[F(R^+)-F(R^-)\big]\,.\end{aligned}$$

The ordered ranking vector $R_{(i)}^{c+t}$ follows indeed the same distribution than the drawn vectors $R_i^{(k)}$. Then:

$$\begin{aligned}F(R^+)-F(R^-) &= \Big(F(R^+)-\widehat{F}_K(R^+)\Big)-\Big(F(R^-)-\widehat{F}_K(R^-)\Big)+\widehat{F}_K(R^+)-\widehat{F}_K(R^-) \\ &\geqslant -2\sup_{\theta\in\mathbb{R}^n}\Big|F(\theta)-\widehat{F}_K(\theta)\Big| + (1-\delta)\,,\end{aligned}$$

as by assumption on the envelope, $\widehat{F}_K(R^+)-\widehat{F}_K(R^-) \geqslant 1-\delta$ almost surely. Then after taking the expectation, we get:

$$\mathbb{P}\Big[\forall i \in [\![n]\!] : R_i^{c+t} \in \Big[R^-_{R_i^c}, R^+_{R_i^c}\Big]\Big] \geqslant 1-\delta-2\mathbb{E}\left[\sup_{\theta}\Big|F(\theta)-\widehat{F}_K(\theta)\Big|\right]\,.$$

It remains to upper bound that expectation. Let us denote $Z = \sup_\theta\Big|F(\theta)-\widehat{F}_K(\theta)\Big|$, let $t \geqslant 0$,

$$\mathbb{E}[Z] = \mathbb{E}[Z\mathbf{1}_{Z\leqslant t}+Z\mathbf{1}_{Z>t}] \leqslant t+\mathbb{P}[Z>t] \leqslant t+n(K+1)e^{-2Kt^2}\,.$$

We have used that Z is bounded by 1 and Theorem A.2. We conclude by choosing $t = \sqrt{\frac{\log nK}{K}}$. □

B Alternative targets

We detail in this section the guarantees we get on the alternative target sets evoked in Section 3.3.

Rank among the test points. A user might be interested in the ranking of the test points among themselves, rather than within the entire dataset. Such a set can be constructed directly from the previous sets, as presented in following Corollary B.1.

Corollary B.1. *For $i \in [\![m]\!]$, let*

$$a_{n+i} = \min \widehat{\mathcal{C}}_k(X_{n+i}), \quad \textit{and} \quad b_{n+i} = \max \widehat{\mathcal{C}}_k(X_{n+i}), \tag{18}$$

$$N^+_{n+i} = \left|\left\{j \in [\![n]\!], R^+_j \leqslant a_{n+i}\right\}\right| \quad \textit{and} \quad N^-_{n+i} = \left|\left\{j \in [\![n]\!], R^-_j \leqslant b_{n+i}\right\}\right|. \tag{19}$$

Then $\widehat{\mathcal{C}}^t_k(X_{n+i}) = [a_{n+i} - N^-_{n+i}, b_{n+i} - N^+_{n+i}]$ is marginally valid, $\mathbb{P}\big[R^t_{n+i} \in \widehat{\mathcal{C}}^t_k(X_{n+i})\big] \geqslant 1 - \alpha$ and has a controlled FCP, with probability $1 - \beta - \delta$:

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\left\{R^t_{n+i} \notin \widehat{\mathcal{C}}^t_k(X_{n+i})\right\} \leqslant \alpha + \lambda_{n,m}\,, \tag{20}$$

with $\lambda_{n,m}$ defined in Proposition 3.8 .

Proof. As in the proof of Theorem 2.1, we assume that the event $E_\delta = \left\{\forall j \in [\![n]\!] : R^{c+t}_j \in [R^-_j, R^+_j]\right\}$ holds. Then if $R^{c+t}_{n+i} \in \widehat{\mathcal{C}}_k(X_{n+i}) \subset [a_{n+i}, b_{n+i}]$ it implies that $R^t_{n+i} \in [a_{n+i} - R^c_{n+i}, b_{n+i} - R^c_{n+i}]$. Thus, we can upper and lower bound R^c_{n+i} by just noticing that $R^c_{n+i} = \left|\left\{j \in [\![n]\!], R^{c+t}_j \leqslant R^{c+t}_{n+i}\right\}\right|$. It follows:

$$N^+_{n+i} \leqslant \left|\left\{j \in [\![n]\!], R^{c+t}_j \leqslant a_{n+i}\right\}\right| \leqslant R^c_{n+i} \leqslant \left|\left\{j \in [\![n]\!], R^{c+t}_j \leqslant b_{n+i}\right\}\right| = N^-_{n+i}\,,$$

as $R^{c+t}_j \in [R^-_j, R^+_j]$. It gives the marginal coverage as, then:

$$[a_{n+i} - R^c_{n+i}, b_{n+i} - R^c_{n+i}] \subset [a_{n+i} - N^-_{n+i}, b_{n+i} - N^+_{n+i}].$$

The control of the FCP is obtained in the same manner using Proposition 3.8. □

Top-K items. It is also possible to target the top-K items by selecting all items whose prediction set includes at least one rank smaller than or equal to K. Corollary B.2 exhibits that this set contains in average $\mathtt{K} - \alpha m$ of the top K items. This can be sufficient if we target a "top" proportion of the items. This strategy is used in Section E.1 to identify top-1 candidates in the Yummy Food dataset.

Corollary B.2. *For $\mathtt{K} \in [\![m]\!]$, let $\widehat{\mathcal{C}}^{\,topK} = \left\{X_{n+i} : \exists \ell \leqslant \mathtt{K}, \ell \in \widehat{\mathcal{C}}_k(X_{n+i})\right\}$. Then*

$$\mathbb{E}\left[\left|\widehat{\mathcal{C}}^{\,topK} \cap \mathcal{C}^{\,topK}\right|\right] \geqslant \mathtt{K} - \alpha m, \tag{21}$$

where $\mathcal{C}^{\,topK} = \left\{X_{n+i} : R^{c+t}_{n+i} \leqslant \mathtt{K}\right\}$ is the set of the true top K items.

Proof. Let us just link the expectation size to the coverage:

$$\begin{aligned}
|(\widehat{\mathcal{C}}^{\texttt{topK}}_k)^c \cap \mathcal{C}^{\texttt{topK}}_k| &= \left|\left\{i \in [\![m]\!] : R^{c+t}_{n+i} \leqslant \mathtt{K},\ \text{and}\ \forall \ell \in [\![\mathtt{K}]\!], \ell \notin \widehat{\mathcal{C}}_k(X_{n+i})\right\}\right| \\
&\leqslant \left|\left\{i \in [\![m]\!] : R^{c+t}_{n+i} \leqslant \mathtt{K},\ \text{and}\ R^{c+t}_{n+i} \notin \widehat{\mathcal{C}}_k(X_{n+i})\right\}\right| \\
&\leqslant \left|\left\{i \in [\![m]\!] : R^{c+t}_{n+i} \notin \widehat{\mathcal{C}}_k(X_{n+i})\right\}\right| \\
&= \sum_{i=1}^{m} \mathbf{1}\left\{R^{c+t}_{n+i} \notin \widehat{\mathcal{C}}_k(X_{n+i})\right\}.
\end{aligned}$$

By taking the expectation, we get that $\mathbb{E}\Big[\ |(\widehat{\mathcal{C}}^{\texttt{topK}}_k)^c \cap \mathcal{C}^{\texttt{topK}}_k|\Big] \leqslant \alpha m$ and then $\mathbb{E}\Big[\ |\widehat{\mathcal{C}}^{\texttt{topK}}_k \cap \mathcal{C}^{\texttt{topK}}_k|\Big] \geqslant \mathtt{K} - \alpha m$. □

C Details on the procedure

C.1 Quantile envelope

Algorithm 2 explicitly details the construction of the quantile envelope, estimated by Monte Carlo with a sample of size K.

Algorithm 2 Quantile envelope procedure

```
**Input:** $n, m, K$, and $\delta \in (0,1)$, $(R_i^c)_{i \in [\![n]\!]}$
**for** $k = 1, \dots, K$ **do**
    $\tilde{R}^{(k)} \leftarrow$ Output of **Algorithm 1**
**end for**
$\widehat{\gamma} \leftarrow \max \gamma$
such that: $\sum_k \mathbf{1}\Big\{\exists i : \tilde{R}_i^{(k)} \notin \Big[\widehat{\mathrm{Q}}_\gamma\Big((\tilde{R}_i^{(\ell)})_\ell\Big), \widehat{\mathrm{Q}}_{1-\gamma}\Big((\tilde{R}_i^{(\ell)})_\ell\Big)\Big]\Big\} \leqslant K\delta$
**for** $j = 1, \dots, n$ **do**
    $\widehat{R}_j^- \leftarrow \widehat{\mathrm{Q}}_{\widehat{\gamma}}\Big((\tilde{R}_j^{(\ell)})_{\ell \in [\![K]\!]}\Big)$
    $\widehat{R}_j^+ \leftarrow \widehat{\mathrm{Q}}_{1-\widehat{\gamma}}\Big((\tilde{R}_j^{(\ell)})_{\ell \in [\![K]\!]}\Big)$
**end for**
**Output:** $\widehat{R}^-_{R_i^c}, \widehat{R}^+_{R_i^c}$ for $i \in [\![n]\!]$.
```

In this algorithm, for $x \in \mathbb{R}^K$ and $u \in (0,1)$, $\widehat{Q}_u(x)$ denotes the empirical quantile of order u of the sample x.

C.2 FCP control

In this section, we give details on the procedure to control the FCP at level $\bar{\alpha} \in (0,1)$. First, let us remark that FCP $:= m^{-1} \sum_{i=1}^m \mathbf{1}\{p_i \leqslant t\}$, where the p_i are given in Eq. (12). In other words, the FCP is the proportion of conformal p-values that are lower than t. Furthermore, we have that [Gazin et al., 2024]:

$$\{\text{FCP} \leqslant \bar{\alpha}\} = \left\{ m^{-1} \sum_{i=1}^m \mathbf{1}\{p_i \leqslant t\} \leqslant \bar{\alpha} \right\} = \left\{ \sum_{i=1}^m \mathbf{1}\{p_i \leqslant t\} \leqslant \lfloor m \cdot \bar{\alpha} \rfloor \right\} = \left\{ p_{(\lfloor m \cdot \bar{\alpha} \rfloor + 1)} > t \right\},$$

where $p_{(a)}$ denotes the a-th smallest value among $(p_1, \dots, p_m)$. We see that controlling the FCP at level $\bar{\alpha}$ is equivalent to control that the ordered p-value $p_{(\lfloor m \cdot \bar{\alpha} \rfloor + 1)}$ is strictly larger than t. Hence, for $\bar{\beta} > 0$, if we denote by $t_{\bar{\beta}}$ the quantile of level $\bar{\beta}$ of the distribution of $p_{(\lfloor m \cdot \bar{\alpha} \rfloor + 1)}$, then $\mathbb{P}(\text{FCP} \leqslant \bar{\alpha}) = \mathbb{P}\big(p_{(\lfloor m \cdot \bar{\alpha} \rfloor + 1)} > t_{\bar{\beta}}\big) \geqslant 1 - \bar{\beta}$.

It remains to find $t_{\bar{\beta}}$. As proven by Gazin et al. [2024, Proposition 2.1], if the scores $V_1, \dots, V_{n+m}$ are exchangeable random variables having no ties almost surely, the vector of conformal p-values $(p_1, \dots, p_m)$ follows a known universal distribution. Hence, it is possible to simulate samples following the same distribution as $p_{(\lfloor m \cdot \bar{\alpha} \rfloor + 1)}$ by first simulating the vector $(p_1, \dots, p_m)$ and then by tacking the $(\lfloor m \cdot \bar{\alpha} \rfloor + 1)$ smallest value. Then, we compute the empirical quantile of order $\bar{\beta}$ to find $t_{\bar{\beta}}$. The full procedure is given in Algorithm 3.

D Additional results on synthetic data

We present in this section additional experiments on the synthetic data of Section 5.1 and on another synthetic data set especially constructed to highlight the adaptivity of the score $s^{\mathbf{VA}}$.

D.1 Additional results for synthetic data of Section 5.1

We complete the results of Section 5.1 when instead of using RankNet we use:

Algorithm 3 Control of the FCP

```
Input: $\bar{\alpha}, \bar{\beta}, n, m$ and $K$
$\tilde{P} \leftarrow$ vector of size $K$
for $k = 1, \dots, K$ do
    Draw $n + m$ uniform random variable $U_i$ on $[0, 1]$
    for $i = 1, \dots, m$ do
        $\tilde{p}_i \leftarrow \frac{1}{m+1}\left(1 + \sum_{j=1}^{m} \mathbf{1}\{U_{j+n} \geqslant U_i\}\right)$
    end for
    $\tilde{P}_k \leftarrow \widehat{\mathrm{Q}}_{\lfloor m \cdot \bar{\alpha} \rfloor + 1}(\tilde{p}_1, \dots, \tilde{p}_m)$
end for
$\widehat{t}_{\bar{\beta}} \leftarrow \widehat{\mathrm{Q}}_{\bar{\beta}}(\tilde{P})$
Output: $\widehat{t}_{\bar{\beta}}$
```

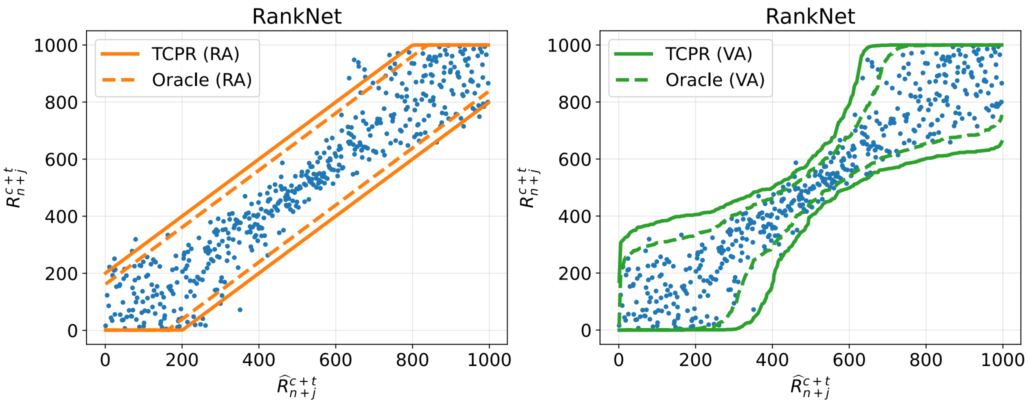


Figure 5: Synthetic data: True ranks R^{c+t}_{n+j} in function of their predicted rank $\widehat{R}^{c+t}_{n+j}$ by RankNet and their prediction sets with scores $s^{\mathbf{RA}}$ and $s^{\mathbf{VA}}$ for $n = m = 500$.

- LambdaMART with 100 trees of 5 leaves trained on $n_{tr} = 300$ data points. Performances, in term of FCP and relative length, for different values of $n \in \{100, 500, 2500\}$ and $m \in \{100, 500, 2500\}$ are presented in Figure 9.
- RankSVM trained with $n_{tr} = 300$. Performances for different values of $n \in \{100, 500, 2500\}$ and $m \in \{100, 500, 2500\}$ are presented in Figure 10.
- BRE applied with 1% of all the possible comparisons. Performances for different values of $n \in \{100, 500, 2500\}$ and $m \in \{100, 500, 2500\}$ are presented in Figure 11. Note that BRE is of different nature than the three other algorithms as it does not learn a score from their features but ranks items using a small number of pairwise comparisons. The ranking errors are due to this limited number of comparisons.

We also provide performances when using RankNet for $n \in \{100, 500, 2500\}$ and $m \in \{100, 2500\}$ in Figure 12. Note that in all the synthetic experiments, we use RankNet with a ReLU Neural Network (NN) of 5 hidden layers of size 10. This NN is trained using `Pytorch` [Paszke, 2019]. Figure 5 displays an example of the prediction sets we construct with TCPR and with the oracle method. All the numerical results are postponed to Section E.3 for clarity of presentation.

Results: As already observed in the main paper, our method, TCPR, always control the false coverage proportion at level $\alpha = 0.1$. Furthermore, at any given size m, as the size n of the calibration set increases, our methods return better sets with FCP closer to 0.1 and increasingly smaller sizes. In general, we need more calibration points than the number of test points to be able to construct prediction sets that are not too large, which is quite intuitive. For n large relatively to m, we reach the performance of the oracle. For LambdaMART and RankSVM, the score function $s^{\mathbf{RA}}$ gives in general better prediction sets than those constructed with $s^{\mathbf{VA}}$. However, this difference is less notable for BRE. Nevertheless, it should be noted that sets with $s^{\mathbf{RA}}$ are less adaptive than the

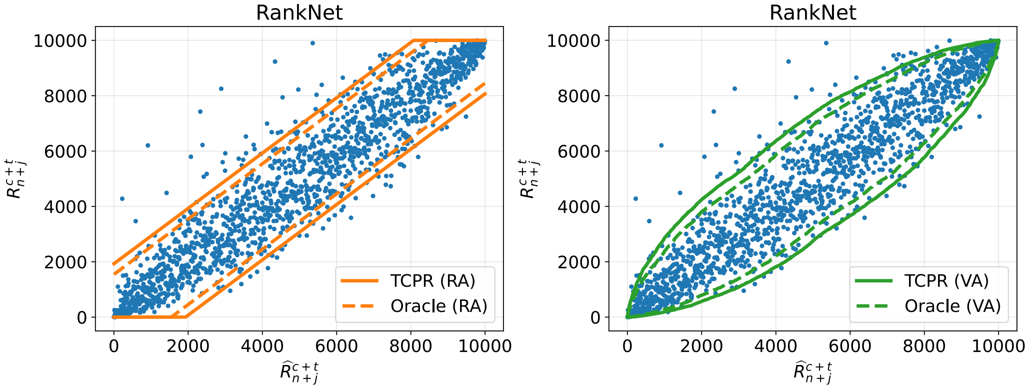


Figure 6: Yummly-10k: True ranks R^{c+t}_{n+j} in function of their predicted rank $\widehat{R}^{c+t}_{n+j}$ by RankNet and their prediction sets with scores $s^{\mathbf{RA}}$ and $s^{\mathbf{VA}}$.

ones with $s^{\mathbf{VA}}$ as explain in Section 5.1. Overall, the difference of performance between the score functions depends on the distribution of the data and the algorithm.

E Additional results on real data

In this section, we present additional information and results on the two real data sets considered in the main paper.

E.1 Yummly-10k

Hyper-parameters: For all the experiments on this data set, we use RankNet with a ReLU NN of 5 hidden layers of size 10. This NN is trained using PyTorch [Paszke, 2019].

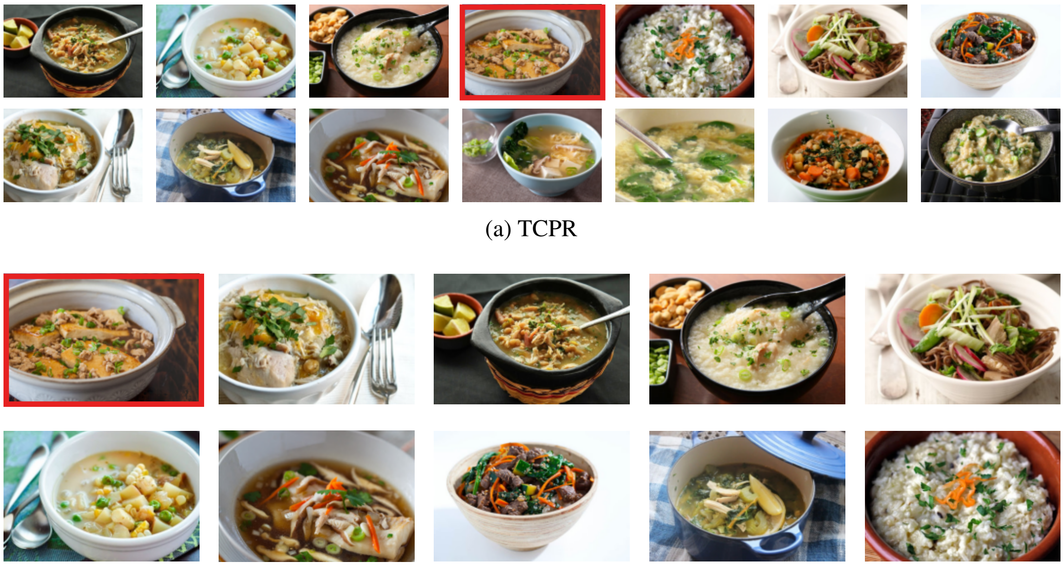
Additional results: Figure 6 illustrates an example of the prediction sets returned by our method and by the oracle. As already seen from Table 1, the differences between TCPR and the oracle are relatively small. This can be explained by the large size of the data set which makes the impact of the envelopes negligible. Notice that from these predictions sets, we can infer who the potential candidates of rank 1 would be. Figure 7 shows these candidates when the prediction sets are constructed using TCPR or the oracle method with $s^{\mathbf{VA}}$. The candidates are all the images containing the rank 1 in their prediction set, i.e. we display $\left\{\text{dishes X} : 1 \in \widehat{\mathcal{C}}(X)\right\}$. As the prediction set of the image of true rank 1 contains 1 with high probability, this set will contain this image with high probability. The image highlighted in red is the randomly chosen reference x^* which is, by definition, the true rank 1 dish (see Section 5.2). Firstly, we observe that the reference x^* is included among these top-1 candidates and that the other dishes are closely related to it. Secondly, as expected, the number of candidates is greater for TCPR (14 candidates) than for Oracle (10 candidates) but not by much. Thirdly, remark that, due to the use of $s^{\mathbf{VA}}$, the prediction sets are narrower for the lower ranks, so the number of candidates remains relatively low. For this specific task, the score $s^{\mathbf{VA}}$ is particularly more adapted than $s^{\mathbf{RA}}$.

E.2 Anime recommendation LTR data set

Data set: In details, this data set consists in the three following lists[3]:

1. A list of 16681 movies with the following associated features: name, genres, is_tv, year, is_adult, above_five_star_users, above_five_star_ratings, above_five_star_ratio.
2. A list of 15163 users with the following associated characteristics: review_count, user_feature avg_score, user_feature score_stddev, user_feature above_five_star_count, user_feature, above_five_star_ratio.

[3] https://www.kaggle.com/datasets/ransakaravihara/anime-recommendation-ltr-dataset



(a) TCPR

(b) Oracle

Figure 7: Candidates for the rank 1 dish when using TCPR or the oracle methods with $s^{\mathbf{VA}}$. The reference x^* (which is the true rank 1 dish) is highlighted in red.

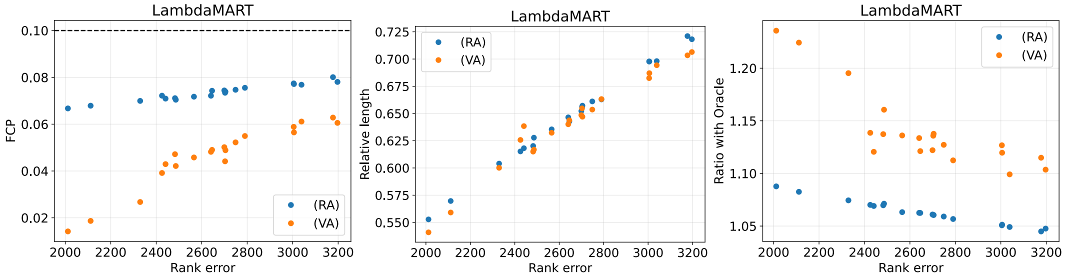


Figure 8: Results on the anime recommendation LTR data set. Each point corresponds to a model LambdaMART(tree,leaves) trees $\in \{50, 100, 200, 300\}$ and leaves $\in \{5, 10, 15, 20\}$. The FCP, relative length, and ratio with the oracle are displayed for each score in function of the average rank error of each model.

3. A list of ratings for 10^6 tuples (movie, user), with values ranging from 0 to 10.

Ranking task and calibration: We use LambdaMART from the LightGBM implementation[4] for this LTR problem. The model assigns a score to each (user, movie) tuple and has been trained on a subset of 15000 users. The reference model has 400 trees and 20 leaves, the smaller ones have trees $= \{50, 100, 200, 300\}$ and leaves $= \{5, 10, 15, 20\}$. For a particular user, we then suppose only having access to the scores predicted by the large model for $n = 2000$ anime, and evaluate the others $m = 16681 - n$ scores with the smaller models.

Results: Figure 8 displays the different metrics with respect to the rank error of all the LambdaMART models. The uncertainty across different instances is well captured by our procedure, as illustrated in the second row of the plots: the size of the prediction intervals increases with the ranking algorithm's errors. For all methods, the FCP is well controlled and remains below the threshold of 0.1. Once again, we observe that the envelope effect is more pronounced for high-performing algorithms: the ratio with the oracle decreases as the ranking error increases. Nevertheless, the ratios remain close to 1 for all methods, suggesting that the overall effect of the envelope is minor.

[4] `https://github.com/microsoft/LightGBM`

E.3 Additional numerical results

This section contains additional experiments for the synthetic dataset (different tuples (n, m)).

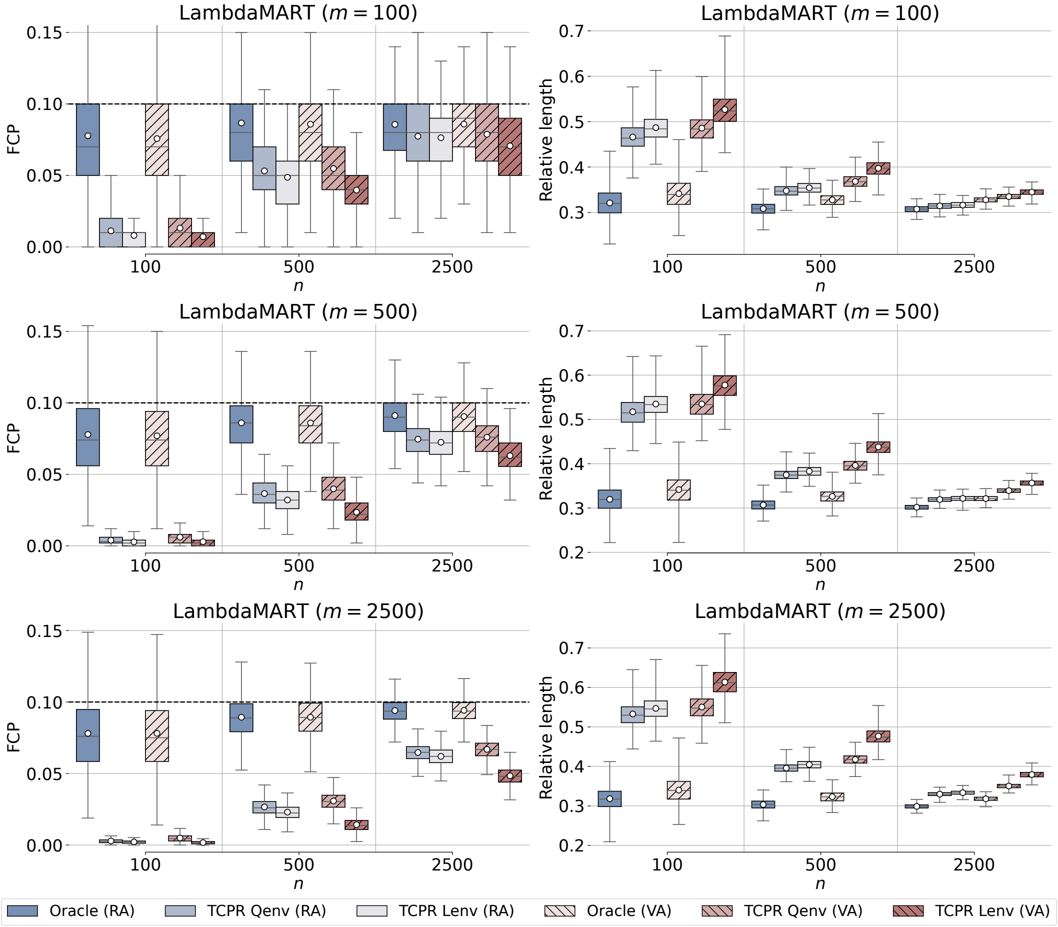


Figure 9: Synthetic data: FCP and relative lengths obtained for LambdaMART with the (**RA**) and (**VA**) score, for the quantile (*Qenv*) and linear (*Lenv*) envelopes when $m = \{100, 500, 2500\}$ and $n \in \{100, 500, 2500\}$. White circles represent the means.

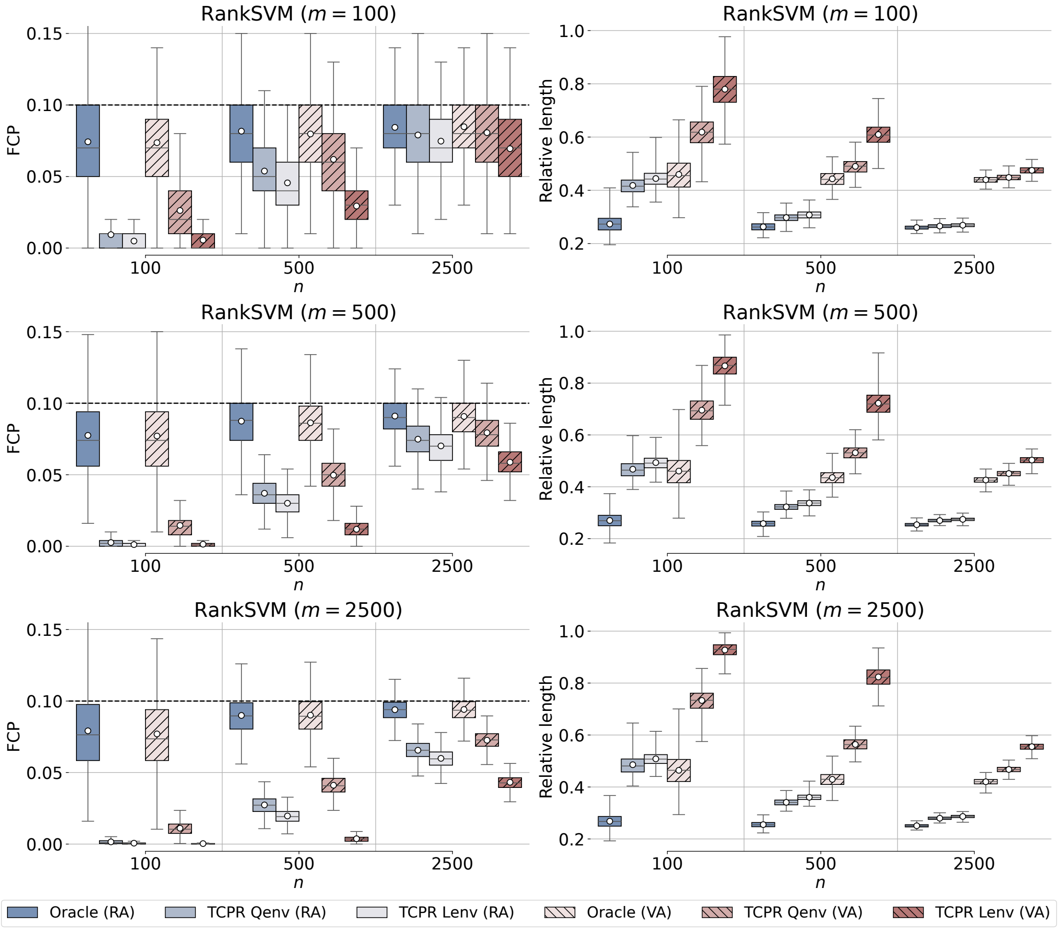


Figure 10: Synthetic data: FCP and relative lengths obtained for RankSVM with the (**RA**) and (**VA**) score, for the quantile (*Qenv*) and linear (*Lenv*) envelopes when $m = \{100, 500, 2500\}$ and $n \in \{100, 500, 2500\}$. White circles represent the means.

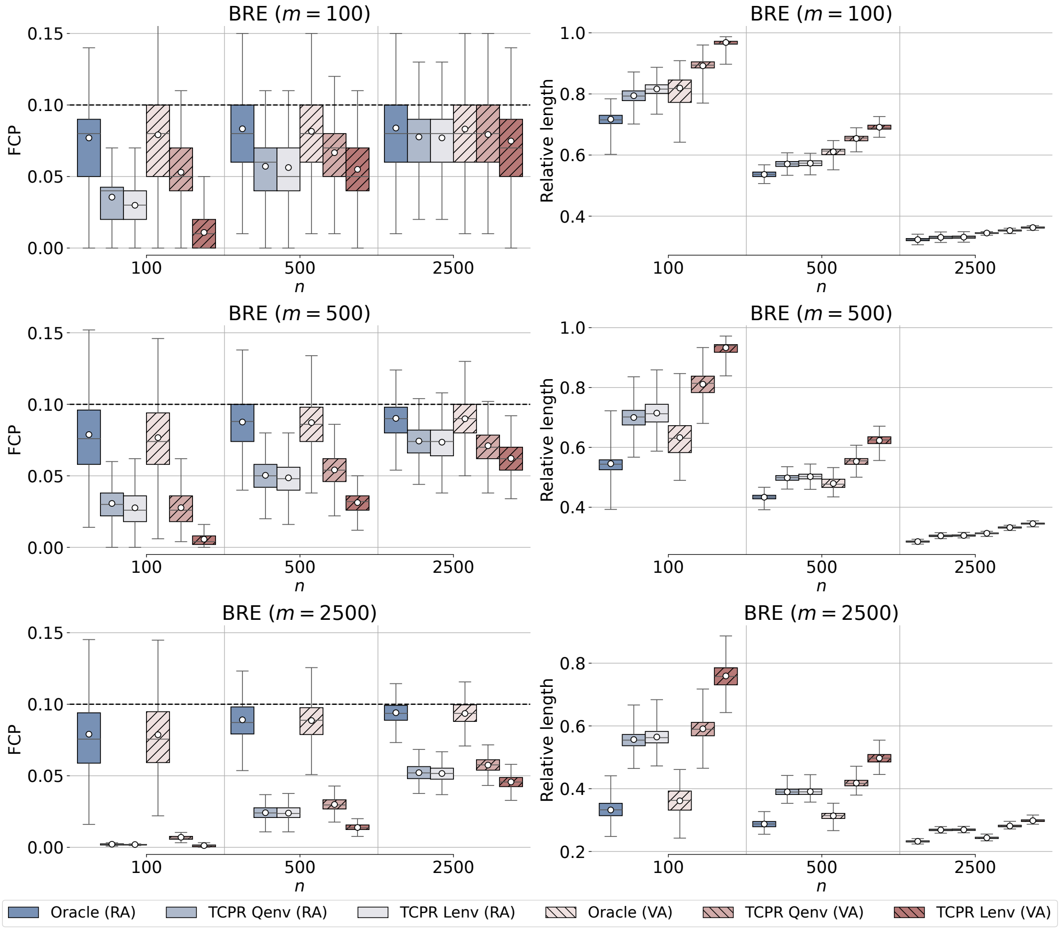


Figure 11: Synthetic data: FCP and relative lengths obtained for BRE with the (**RA**) and (**VA**) score, for the quantile (*Qenv*) and linear (*Lenv*) envelopes when $m = \{100, 500, 2500\}$ and $n \in \{100, 500, 2500\}$. White circles represent the means.

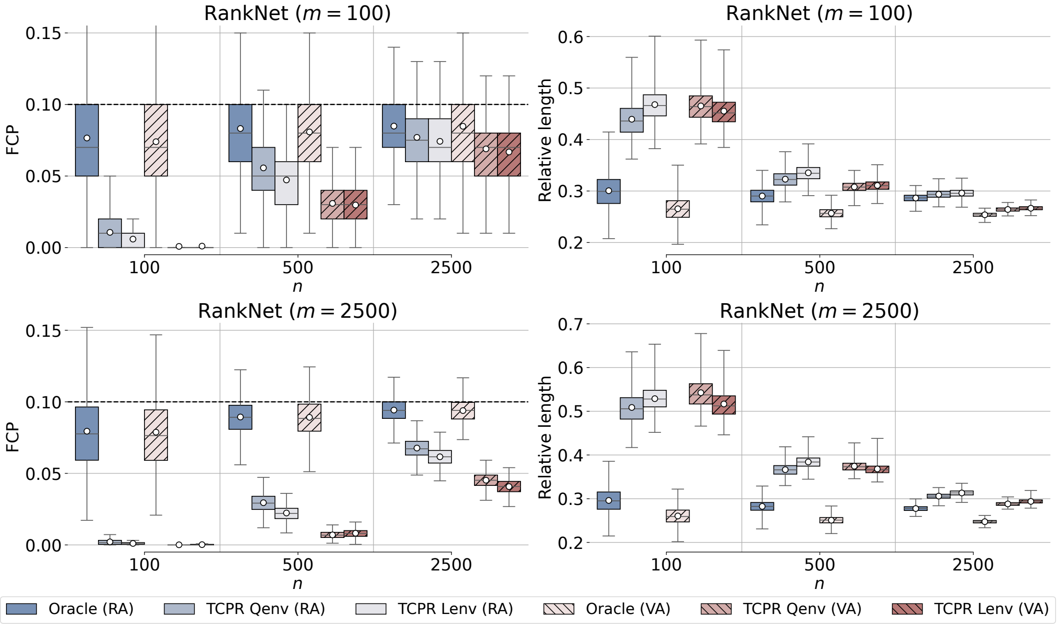


Figure 12: Synthetic data: FCP and relative lengths obtained for RankNet with the (**RA**) and (**VA**) score, for the quantile (*Qenv*) and linear (*Lenv*) envelopes when $m = \{100, 2500\}$ and $n \in \{100, 500, 2500\}$. White circles represent the means.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and the introduction explain our approach et the setting of full ranking we consider.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: In all the experiments, we evaluate the ratio with the oracle length and discuss multiple times the potentially disruptive impact of the envelope when assessing the performance of a high-performing algorithm.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proofs are given in Appendix A. The assumptions are stated in the main paper and recall at each result.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

Justification: The Python code is provided in the supplementary material, and the datasets are either simulated or open data. Overall the method is simple and could be reimplemented from scratch by an interested reader.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The Python code is provided in the supplementary material, and the datasets are either simulated or open data. Overall the method is simple and could be reimplemented from scratch by an interested reader.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: These informations are in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We provide boxplots and the standard deviations of all the metrics.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

Justification: Our experiments do not require a lot of computational resources and run on a standard machine.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We are fully in accordance with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: It is discussed in the introduction.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: This is not applicable to our method.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the datasets we used precisely. For the code, apart from standard Python code (numpy, pytorch, matplotlib, etc.) we rely mostly on our own code. It will be release under MIT License.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: A notebook is provided with our code in the supplementary material.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not have experiments with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We do not have experiments with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: We do not use LLMs.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.