Risk Optimization for Learning to Rank through Portfolio Theory

Oscar Rolando Ramirez Milian
o.r.ramirezmilian@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Harrie Oosterhuis
h.oosterhuis@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Abstract

Learning-to-rank (LTR) methods concern the optimization of models that rank items based on their features, where existing methods assume relevance labels are known and determinate. However, in many scenarios relevancy is unknown, e.g., when learning from user interactions, or stochastic, e.g., when user queries are ambiguous. But even when provided a distribution of label uncertainty, existing LTR methods cannot account for the uncertainty since they only optimize mean ranking performance which reduces uncertainty distributions to pointwise mean predictions. As a result, current LTR can produce very risky behaviors that increase mean outcomes while still giving very negative outcomes significant probability. For example, a risky ranking for an ambiguous query would be to only provide items related to only possible interpretation, thus risking that the user does not like any items, whereas a safe ranking would be diverse so that users find at least one relevant item in most probable interpretations.

Previous work has proposed methods for constructing a single ranking using portfolio theory to deal with this issue. We take this approach even further and apply portfolio theory to LTR losses, thereby, extending it to the optimization of ranking models. Since there is no single objective concept of *risk*, this work first lays out the different risk measures established in modern portfolio theory, subsequently, we translate these to LTR losses that can be computed or estimated in practical settings. Our key contribution is a single versatile framework that can be used for optimize a wide variety of risk measures for LTR. Our experiments reveal that our risk-aware losses result is safer and more robust ranking models.

CCS Concepts

• **Information systems → Presentation of retrieval results**.

Keywords

Learning to Rank, Reinforcement learning, Portfolio Theory.

1 Introduction

Most learning-to-rank (LTR) methods, assume that a deterministic relevance label is given per item and the standard goal of LTR is to find a model that ranks according to these labels [2–4, 8, 11, 14, 15, 17, 19, 29].[1] In practice, LTR is often applied in settings where relevance is ambiguous or inferred [5, 7, 9, 10, 13, 16, 21, 23, 24]. User intent is only imperfectly expressed by queries, making relevance inherently uncertain, and explicit relevance labels are frequently unavailable or too costly, so they are inferred from noisy, biased, and often sparse interaction data [8, 9, 28]. The resulting inferences are thus never certain due to noise and bias in user interactions or because little to no interactions are available for a query-item combination [7, 10, 13, 16, 28]. In a situation where a model of uncertainty is available in the form of a distribution of relevance [20, 25], one could simply change the existing LTR optimization goal to maximize the expected utility under that distribution. Yet for standard ranking metrics/losses, the optimal policy would then simply be a deterministic ranking model that ranks according to the expected relevance. This approach would not account for any of the risks stemming from the uncertainty, as they only consider the mean case. Consider a simple example, take an ambiguous query with three possible intents where 35% of users have intent A, 33% B, and 32% C, and assume that no document can be relevant for multiple intents. To maximize the expected precision@5 under this distribution, one should only show documents relevant to intent A, i.e., $[A_1, A_2, A_3, A_4, A_5]$ results in an expected precision of 0.35. However, this is clearly a very risky approach because when a user shows up from group B or C, which happens in 65% of cases, then precision@5 is 0. In contrast, a more diverse ranking such as $[A_1, B_2, C_3, A_1, B_2]$ leads to a slightly lower expected precision@5 of 0.336 but has at least one relevant document for all users. This connection between risk minimization and diversification was recognized by Wang and Zhu [26] and subsequent work [22, 24, 27], who proposed greedy, portfolio-theoretic algorithms that construct risk-averse (more diverse) rankings as an additional inference step in the retrieval pipeline. These methods, however, are typically computationally costly at inference time and are not formulated as LTR objectives; they do not directly learn risk-aware ranking models. Importantly, they have not made the connection with the LTR field as they are not concerned with model optimization. Our work addresses this gap directly by translating modern portfolio risk measures into differentiable LTR loss functions via exposure based methods. We propose a framework for translating these measures into novel risk-aware LTR loss function estimators that can be optimized directly through exposure-based LTR. Our results demonstrate that our framework is capable of learn risk-aware ranking models for each of our proposed risk measures. As a result, our framework can yield policies that are robust to uncertainty, such that, they provide improved effectiveness in most possible scenarios.

[1] This is even true for existing probabilistic LTR methods [4, 14, 15, 17] as there is guaranteed to exist a deterministic ranking model that maximizes the expected utility when relevance labels are given and deterministic.

CONSEQUENCES '26, Minneapolis, United States

2 Related Work: Portfolio Theory & Risks in IR

Risk management is typically formalized via portfolio optimization, where a limited budget is allocated across assets to trade off expected return and risk. Classical approaches focus on maximizing expected return, but modern portfolio theory emphasizes diversification to reduce risk without excessively sacrificing performance. Markowitz [12] showed that optimizing expected return alone may concentrate all wealth in a single asset and proposed the mean-variance framework, in which portfolios are evaluated jointly by their expected return and the variance of returns, and efficient portfolios minimize variance for a given expectation. Subsequent work has developed more refined ways to quantify risk, noting that variance penalizes upside and downside symmetrically while investors are typically more concerned with large losses. This motivated *downside* and *tail risk* measures. Wang and Zhu [26] were the first to propose applying modern portfolio theory from economics to document rankings for information retrieval (IR). In their foundational work [26], they introduced a mean-variance ranking algorithm that models document relevance as a random variable, estimates variances and covariances between documents, and then constructs rankings that trade off expected gain against variance by selecting a diversified set of documents that maximizes a portfolio-style utility over the entire ranked list. Later on, Wang et al. [27], Silva Rodrigues et al. [22] and Takehi et al. [24] built on these ideas to improve learning and develop more robust evaluation methods.

3 Background: Portfolio Risk Functionals

Let X denote the utility of a portfolio of stocks to be optimized, with higher values being preferable,[2] the classical approach would optimize the portfolio to maximize the expected value $\mathbb{E}\big[X\big]$. However, the expected value can be skewed by a single possible outcome with a low probability but extremely high value. Thus, a high expected value does not guarantee a high probability for utility being high, in contrast, it is very much possible that in the large majority of possibility, the utility is extremely low. For this reason, modern portfolio theory recognizes maximizing expected utility as a very *risky* approach, instead, it proposes to optimize a trade-off between the expected utility and a risk functional: $\mathbb{E}[X] - \lambda\mathcal{R}[X]$. We consider risky anything that deviates from what is expected. We consider the following functionals: *Variance:* $\mathbb{V}[X] = \mathbb{E}\big[(X - \mathbb{E}[X])^2\big]$. This is the most direct measure of uncertainty. However, it quantifies overall dispersion, it does not specifically target bad outcomes, which are most relevant to avoid risk. *Lower Semivariance:* $\mathbb{V}^-[X] = \mathbb{E}\big[(\mathbb{E}[X] - X)^2\mathbb{1}_{X\leq\mathbb{E}[X]}\big]$. In response to the inability of variance to distinguish between favorable and unfavorable deviations from the mean, semivariance considers the *downside risk* by only considering the deviation of outcomes below the expected value. *(Semi-)Standard deviation*: A limitation of variance and semivariance is their usage of the square of the expected difference, which is a different scale than the expected value, thus the standard deviation is also often used: $\sqrt{\mathbb{V}}[X]$, or the semi-standard deviation: $\sqrt{\mathbb{V}^-}[X]$. *Value at Risk.* A difficulty with (semi-)variance is that because they provide a measure or dispersion, they do not provide a directly interpretable indication of risk. This issue is resolved by the value at risk (VaR) functional which is defined as the quantile of the utility distribution, for parameter $\alpha \in (0, 1)$: $\mathcal{V}_\alpha[X] = \inf\{\tau \in \mathbb{R} : \mathrm{P}(X \leq \tau) \geq \alpha\}$. For example, with $\alpha = 0.5$, the value at risk is simply the median utility. Thus, $\mathcal{V}_\alpha$ indicates the utility value that at least $\alpha \cdot 100\%$ of outcomes have. The corresponding risk functional measures the gap between the expected utility and this lower-tail performance: $\mathcal{R}[X] = \mathbb{E}[X] - \mathcal{V}_\alpha[X]$.

[2] Standard portfolio theory uses losses with lower values being preferable, we choose to use utility with higher values being preferable to better match the LTR setting.

4 Exposure-Based Learning to Rank with Stochastic Relevance Labels

Our setting concerns the optimization of a probabilistic ranking model π that given a query q produces a ranking of K documents: $y = (y_1, \ldots, y_K)$; where $y \sim \pi(q)$ and the probability of y is denoted by $\pi(y \mid q)$. Traditionally, the relevance $\rho_{d|q}$ of a document d w.r.t. a query q, is presumed to be deterministic and known with certainty. The utility of a ranking y is a product of position weights $\theta_k \in \mathbb{R}_{\geq 0}$ and the relevance of the item per position: $U(y \mid q, \rho) = \sum_{k=1}^{K} \theta_k \rho_{y_k|q}$. The utility of a policy π is the expected utility of its ranking: $U(\pi \mid q, \rho) = \mathop{\mathbb{E}}_{y\sim\pi(q)}[U(y \mid q, \rho)]$. We reframe utility using expected exposure per document [17]: $\theta_{d|\pi,q} = \mathop{\mathbb{E}}_{y\sim\pi(q)}[\theta_{\mathrm{rank}(d|y)}]$. We can now rewrite the utility as a weighted sum over documents: $U(\pi \mid q, \rho) = \sum_d \theta_{d|\pi,q}\rho_{d|q}$. The advantage of the exposure-based framing is two-fold: Firstly, exposure-based LTR greatly simplifies the implementation of new loss functions [17] which greatly benefits this work. Secondly, the exposure-based framing matches portfolio theory very well: The documents become analogous to the stock and *exposure per document* become analogous to the portfolio, i.e., how invested policy π is in each of them. Thereby, the exposure $\theta_{d|\pi,q}$, becomes how much of that value is received under π. In contrast with previous LTR work and consistent with the framework introduced in [26], in our setting, the relevance labels are stochastic following a probability law: $\mathrm{P}(\rho \mid q)$. We aim to optimize π considering the risks stemming from the uncertainty under the relevance distribution. We take a portfolio theory approach following Section 3 and optimize a linear combination of the expected utility and a risk functional $\mathcal{R}$ with $\lambda \in [0, 1]$, we call this the *risk-corrected expectation* for the utility:

$$\mathcal{U}_\lambda(\pi \mid q) = \mathbb{E}_\rho\big[U(\pi \mid q, \rho)\big] - \lambda\,\mathcal{R}_\rho\big[U(\pi \mid q, \rho)\big], \qquad (1)$$

where the expectations and the risk functional are always taken with respect to $\mathrm{P}(\rho \mid q)$. Finally, we wish to optimize $\mathcal{U}_\lambda$ over the natural distribution of queries, but we assume that only a sampled set of Q queries is available and we optimize the empirical mean: $\overline{\mathcal{U}}_\lambda(\pi) = \frac{1}{Q}\sum_{i=1}^{Q}\mathcal{U}_\lambda(\pi \mid q_i)$.

5 Proposed risk optimization algorithm

The description of our main algorithm is accompanied by a visual overview in Figure 1. Our algorithm takes the following as input: (1) the tradeoff parameter $\lambda \in [0, 1]$; (2) a set of M documents represented by corresponding document features; (3) a stochastic ranking model π to be optimized; (4) a function from which relevances $\rho_{d|q}$ can be sampled based on features; (5) the parameter N for the number of samples to take; and (6) the risk functional $\mathcal{R}$.

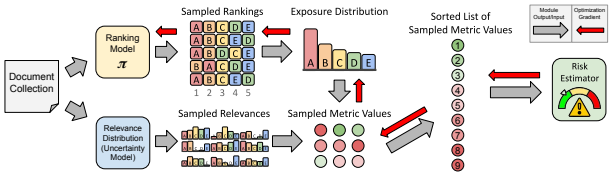

Figure 1: Schematic overview of our general risk optimization framework for LTR. Rankings are sampled from the stochastic ranking model π in order to estimate the exposure distribution. Relevances are sampled from the relevance distribution.

We start by estimating the exposure per document $\theta_{d|\pi,q}$ based on π and the available features, we refer to [17] for an efficient estimator of exposure. Subsequently, N relevance samples are taken from the distribution based on the document features. The utility value for each relevance sample is then computed, this is a simple dot product between a vector of document exposures and a matrix of relevances, resulting in N sampled utility values. The mean of the sampled utilities is then computed, and interpolated with the risk by a factor of λ, culminating in our estimate value of the risk-corrected expected utility. Finally, the ranking model π is optimized through gradient descent on $\mathcal{U}_\lambda(\pi \mid q)$, e.g., using an auto-differentiation framework [1]. Our final estimate is end-to-end differentiable to π as long as the risk functional $\mathcal{R}$ is differentiable, since all other steps fully differentiable. We note that our only assumption about the relevance distribution is that samples can be taken from it. We consider this a very light assumption which means it should function well with a wide class of uncertainty models.

6 Experimental setup

Datasets We perform our experiments on two learning-to-rank datasets: *MSLR-Web-10K* [18] and the *Istella-S* [6]. We optimize for a top-K ranking setting with $K = 10$, and as utility metric, we use DCG@10. That is the weights per rank are: $\theta_k = (\log_2(k+1))^{-1}$, for $0 \leq k \leq 10$ and $\theta_k = 0$ otherwise. *Ranking models.* For our ranking models, we train neural networks with three hidden layers of 256, 128, and 64 units to predict document exposure. We use the exposure-based differentiable distribution presented in [17] trained for 500 epochs. For exposure-based gradient estimation of the policy π, we use $N_\theta = 120$ sampled ranking per estimate. Training uses batches of 2,750 queries and the AdamaxW optimizer with a learning rate of 1×10^{-2}. *Label distributions.* Our setup requires a label distribution that captures the uncertainty in the relevance labels. Since, to the best of our knowledge, no LTR dataset provides uncertainty estimates for its relevance labels, we simulate label distributions for our experiments. Specifically, we cluster all documents into 25 groups using the K-means algorithm with Euclidean distance. For each cluster, we compute the empirical distribution of relevance labels, i.e., the observed label frequencies among the documents in that cluster. Because the correlation between relevance labels can have a substantial impact on risk-aware ranking, we consider an extreme yet realistic scenario, in which relevance labels are perfectly correlated within each cluster. In this setting, a single relevance label is sampled from the empirical label distribution of each cluster, and all query-document pairs associated with documents in that cluster are assigned that label. This scenario models a latent confounding factor that simultaneously affects the relevance of all documents in a cluster. For example, if a user's intent matches a particular topic, then all documents associated with that topic become relevant; otherwise, they all become irrelevant. *Evaluation and baselines.* We compare with two baselines one that optimizes the expected value $\mathbb{E}[U]$, and another that optimizes the expected value plus the entropy of the exposure distribution over documents $\mathbb{E}[U]) + H(\theta)$. This entropy incentives the baseline to increase the diversity of its rankings. Since there is no objective best metric for risk, we evaluate several metrics that indicate different quantiles of outcomes. Specifically, we report the expected DCG@10 (i.e., the mean), and the 50% (the median), 25% and 10% quantiles.[3] To better understand the behavior of our learned ranking models, we also report the diversity of their rankings through: (i) the mean of the ratio of documents that receives 95% of exposure for a query, and (ii) the entropy of the exposure distribution, $H(\theta)$. We take $N = 1000$ label samples, for both gradient estimation in optimization and metric estimation for evaluation. We report average results over ten independent runs.

7 Discussion

The results for MSLR-WEB10K and ISTELLA-S are shown in Table 1. We first examine the baselines (top rows per dataset). Optimizing $\mathbb{E}$ yields the highest expected DCG@10 on MSLR and close to the highest performance on Istella as expected since the objective matches the metric. However, the quantiles drop substantially: on both datasets the gap between mean and quantiles is substantial. This is especially pronounced on Istella where there is a more than 10% chance that performance is dramatically low, a clear indication of there is a substantial risk in optimizing expected utility by itself. The diversification baseline $\mathbb{E} + H(\theta)$ behaves differently across datasets. On MSLR, it increases diversity but reduces the expected DCG@10 by over 15%; on Istella, it increases diversity without harming mean performance, showing that higher diversity need not result in average effectiveness. In both datasets, the 25% and 10% quantiles improve over $\mathbb{E}$, suggesting that diversification can reduce risk. However, some of the risk functionals have lower or comparable values for the diversity metrics, yet reach higher performance in the quantiles. This appears to indicate that diversification does not directly minimize risks, but instead acts as a proxy. An advantage of the diversification baseline is that it does not require information about the relevance label distribution beyond the mean values, yet this also means that it is unable to directly minimize risks stemming from that distribution. Conceptually, maximizing diversity is suboptimal for risk optimization when there are many documents which are never relevant, and thus, assigning exposure to them increases diversity but does not decrease any risk. Turning to variance-based objectives. The semi-standard deviation appears to provide the most robust performance, in all settings it leads to improved 25% and 10% quantile performance, and mantains comparable medians to $\mathbb{E}$; when $\lambda = 1$, these improvements come at a reduction of mean performance of less than 4%. In contrast

[3]We choose not to normalize the DCG since the normalization factor drastically varies with the sampled relevances, making the mean normalized DCG difficult to interpret.

Table 1: Performance of variance-based risk functionals on MSLR-WEB10K and Istella-S. We compare variance $\mathbb{V}$, std. dev $\sqrt{\mathbb{V}}$, semivariance $\mathbb{V}^-$, semi-std. dev. $\sqrt{\mathbb{V}^-}$, and VaR $\mathcal{V}$ with interpolation factors $\lambda = 0.5$ and $\lambda = 1$. Metrics: DCG@10 mean, 50/25/10% quantiles; average minimum number of documents that receive 95% of exposure per query; and average entropy of the exposure per query. For DCG@10 we report absolute values and percentages relative to the expected-utility baseline. Bold indicates the best, underlined the second-best value per metric and setting. Results are average ± std. dev. over ten runs.

	Risk Functional		$\mathbb{E}$		Q50		Q25		Q10		Docs. 95% exp.	$H(\theta)$
MSLR-WEB 10K	$\mathbb{E}$		**5.63 ± 0.03**	**100% ± 0%**	4.00 ± 0.02	100% ± 0%	1.79 ± 0.02	100% ± 0%	0.68 ± 0.02	100% ± 0%	0.14 ± 0.00	2.36 ± 0.02
	$\mathbb{E} + H(\theta)$		4.68 ± 0.07	83% ± 1%	3.78 ± 0.06	94% ± 2%	2.37 ± 0.04	132% ± 3%	1.45 ± 0.02	214% ± 7%	0.80 ± 0.02	4.30 ± 0.04
	$\mathbb{V}$	$\lambda = 1/2$	4.10 ± 0.04	73% ± 1%	3.33 ± 0.05	83% ± 2%	2.14 ± 0.03	120% ± 2%	1.36 ± 0.02	202% ± 3%	0.93 ± 0.01	**4.55 ± 0.01**
	$\mathbb{V}^-$	$\lambda = 1/2$	4.67 ± 0.07	83% ± 1%	3.81 ± 0.06	96% ± 1%	**2.42 ± 0.03**	**135% ± 2%**	1.49 ± 0.02	220% ± 5%	0.72 ± 0.06	4.11 ± 0.09
	$\sqrt{\mathbb{V}}$	$\lambda = 1/2$	5.10 ± 0.10	92% ± 1%	4.03 ± 0.03	101% ± 1%	2.36 ± 0.10	129% ± 6%	1.27 ± 0.15	179% ± 21%	0.43 ± 0.05	3.40 ± 0.16
	$\sqrt{\mathbb{V}^-}$	$\lambda = 1/2$	5.41 ± 0.02	96% ± 1%	4.05 ± 0.04	101% ± 1%	2.16 ± 0.08	122% ± 5%	0.97 ± 0.05	148% ± 10%	0.22 ± 0.02	2.72 ± 0.06
	$\mathbb{V}$	$\lambda = 1$	4.10 ± 0.04	73% ± 1%	3.33 ± 0.05	83% ± 2%	2.14 ± 0.03	120% ± 2%	1.36 ± 0.02	202% ± 3%	**0.93 ± 0.01**	4.55 ± 0.01
	$\mathbb{V}^-$	$\lambda = 1$	4.24 ± 0.06	75% ± 1%	3.51 ± 0.07	87% ± 1%	2.28 ± 0.04	127% ± 2%	1.47 ± 0.03	216% ± 7%	0.86 ± 0.03	4.41 ± 0.05
	$\sqrt{\mathbb{V}}$	$\lambda = 1$	4.55 ± 0.03	81% ± 0%	3.75 ± 0.02	94% ± 1%	2.40 ± 0.03	134% ± 2%	**1.51 ± 0.04**	**224% ± 9%**	0.78 ± 0.04	4.23 ± 0.07
	$\sqrt{\mathbb{V}^-}$	$\lambda = 1$	5.09 ± 0.06	90% ± 2%	3.99 ± 0.03	100% ± 1%	2.35 ± 0.03	131% ± 2%	1.27 ± 0.06	188% ± 8%	0.42 ± 0.06	3.39 ± 0.11
	$\mathcal{V}_{0.50}$	$\lambda = 1$	5.31 ± 0.06	95% ± 1%	**4.09 ± 0.04**	**102% ± 1%**	2.22 ± 0.10	123% ± 6%	1.09 ± 0.11	158% ± 17%	0.26 ± 0.07	2.89 ± 0.20
	$\mathcal{V}_{0.25}$	$\lambda = 1$	4.94 ± 0.12	88% ± 2%	3.93 ± 0.07	98% ± 2%	2.41 ± 0.03	134% ± 2%	1.36 ± 0.06	203% ± 9%	0.59 ± 0.06	3.75 ± 0.13
	$\mathcal{V}_{0.10}$	$\lambda = 1$	4.65 ± 0.08	83% ± 2%	3.76 ± 0.07	94% ± 2%	2.38 ± 0.04	133% ± 3%	1.46 ± 0.04	215% ± 10%	0.70 ± 0.06	4.07 ± 0.10
ISTELLA-S	$\mathbb{E}$		11.27 ± 0.05	100% ± 0%	5.41 ± 0.01	100% ± 0%	0.37 ± 0.01	100% ± 0%	0.00 ± 0.00	100% ± 0%	0.13 ± 0.00	2.35 ± 0.01
	$\mathbb{E} + H(\theta)$		11.15 ± 0.03	99% ± 0%	5.37 ± 0.01	99% ± 0%	0.45 ± 0.02	123% ± 6%	0.00 ± 0.00	2K% ± 924%	0.25 ± 0.02	2.73 ± 0.03
	$\mathbb{V}$	$\lambda = 1/2$	3.01 ± 1.54	27% ± 14%	1.21 ± 0.93	22% ± 17%	0.16 ± 0.12	43% ± 34%	0.00 ± 0.00	93% ± 158%	0.44 ± 0.10	3.57 ± 0.36
	$\mathbb{V}^-$	$\lambda = 1/2$	3.01 ± 0.28	27% ± 2%	1.77 ± 0.19	33% ± 3%	0.49 ± 0.05	134% ± 14%	0.03 ± 0.00	23K% ± 8K%	**0.78 ± 0.06**	**4.27 ± 0.11**
	$\sqrt{\mathbb{V}}$	$\lambda = 1/2$	10.87 ± 0.06	96% ± 1%	5.54 ± 0.03	102% ± 1%	0.69 ± 0.05	188% ± 13%	0.00 ± 0.00	1K% ± 800%	0.17 ± 0.02	2.50 ± 0.05
	$\sqrt{\mathbb{V}^-}$	$\lambda = 1/2$	**11.29 ± 0.01**	**100% ± 0%**	5.43 ± 0.01	100% ± 0%	0.38 ± 0.01	102% ± 2%	0.00 ± 0.00	117% ± 31%	0.14 ± 0.00	2.36 ± 0.02
	$\mathbb{V}$	$\lambda = 1$	3.01 ± 1.54	27% ± 14%	1.21 ± 0.93	22% ± 17%	0.16 ± 0.12	43% ± 34%	0.00 ± 0.00	93% ± 158%	0.44 ± 0.10	3.57 ± 0.36
	$\mathbb{V}^-$	$\lambda = 1$	1.54 ± 0.09	14% ± 1%	0.83 ± 0.06	15% ± 1%	0.23 ± 0.02	62% ± 4%	0.02 ± 0.00	11K% ± 4K%	0.71 ± 0.03	4.14 ± 0.06
	$\sqrt{\mathbb{V}}$	$\lambda = 1$	3.04 ± 0.60	27% ± 5%	1.84 ± 0.37	34% ± 7%	0.50 ± 0.09	136% ± 26%	0.03 ± 0.00	21K% ± 8K%	0.73 ± 0.09	4.06 ± 0.21
	$\sqrt{\mathbb{V}^-}$	$\lambda = 1$	11.04 ± 0.04	98% ± 0%	5.52 ± 0.02	102% ± 0%	0.66 ± 0.06	178% ± 18%	0.00 ± 0.00	1K% ± 405%	0.18 ± 0.01	2.54 ± 0.02
	$\mathcal{V}_{0.50}$	$\lambda = 1$	10.85 ± 0.08	96% ± 1%	**5.59 ± 0.01**	**103% ± 0%**	0.60 ± 0.05	162% ± 15%	0.00 ± 0.00	674% ± 212%	0.14 ± 0.00	2.37 ± 0.01
	$\mathcal{V}_{0.25}$	$\lambda = 1$	8.48 ± 0.23	75% ± 2%	4.92 ± 0.10	91% ± 2%	**1.11 ± 0.01**	**302% ± 4%**	0.03 ± 0.00	21K% ± 7K%	0.33 ± 0.01	3.17 ± 0.05
	$\mathcal{V}_{0.10}$	$\lambda = 1$	5.99 ± 0.10	53% ± 1%	3.59 ± 0.06	66% ± 1%	0.94 ± 0.01	256% ± 6%	**0.06 ± 0.00**	**36K% ± 11K%**	0.58 ± 0.03	3.81 ± 0.07

with that baseline, performance drops considerably less when going to the lower quantiles, a clear indication that risks have been better mitigated by this method. Variance appears to be a worse choice as it leads to a decrease in performance for all quantiles on Istella, and for all but the 10% quantile on MSLR in the sampled per document setting. The standard deviation and semi-standard deviation appear to result in higher values for the 50% (on both datasets) and 25% (on Istella) quantiles. The key difference between these methods is the square root, which puts their values on a similar scale as the expected value; this likely explains their better performance. The trade-off between standard and semi-standard deviation appears somewhat unpredictable and is difficult to anticipate without empirically observing the resulting performance. Thus, while variance-based risk measures can mitigate risk, it remains difficult to predict and quantify which risks they will reduce. Finally, we consider the value at risk-based functional. As expected, this functional provide better performance than $\mathbb{E}$ for the quantile that they optimize. Substantial improvement is provided for the 25% and 10% quantiles, indicating that risks are successfully mitigated. Additionally, the value at risk-based methods are the only methods that show consistent improvements in the 10% quantile of the sample per cluster setting.

8 Conclusion

In contrast with previous LTR work, our framework can optimize metrics beyond the expected utility of a ranking model; and in contrast with earlier work on portfolio theory for IR, our framework is the first to optimize ranking models using risk functionals and thus to link portfolio theory to the LTR field. The conceptualization of risk is subjective, but our framework is designed to optimize a variety of risk functionals. In this work, we proposed estimators for optimizing seven different risk functionals within our framework, because our framework builds on exposure-based LTR, their implementation in our framework requires little effort. Our results indicate that our framework leads to more robust ranking models for most risk functionals, which means that their performance degrades less in lower quantiles than when optimizing the expected value. In other words, the probability of low performance is substantially lower when using risk optimization, instead of optimizing mean performance. While there is no clear best risk functional, we found the semi-standard deviation and VaR-based functionals lead to more robustness than using variance as the risk functional. Additionally, we argue that the Value at Risk provides better interpretability, as their optimization objective (including the α parameter) is easier to describe than variance-based risk functionals. Overall, we expect this framework to provide a foundation for future work on incorporating and studying additional risk functionals in learning-to-rank.

References

[1] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, et al. 2018. JAX: composable transformations of Python+ NumPy programs. (2018).

[2] Christopher Burges, Robert Ragno, and Quoc Le. 2006. Learning to Rank with Nonsmooth Cost Functions. In *Advances in Neural Information Processing Systems*, B. Schölkopf, J. Platt, and T. Hoffman (Eds.), Vol. 19. MIT Press. https://proceedings.neurips.cc/paper_files/paper/2006/file/af44c4c56f385c43f2529f9b1b018f6a-Paper.pdf

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

[4] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning* (Corvalis, Oregon, USA) *(ICML '07)*. Association for Computing Machinery, New York, NY, USA, 129–136. doi:10.1145/1273496.1273513

[5] Steve Cronen-Townsend. 2002. Quantifying query ambiguity. In *Proceedings of the second international conference on Human Language Technology Research*.

[6] Domenico Dato, Claudio Lucchese, Franco Maria Nardini, Salvatore Orlando, Raffaele Perego, Nicola Tonellotto, and Rossano Venturini. 2016. Fast Ranking with Additive Ensembles of Oblivious and Non-Oblivious Regression Trees. *ACM Transactions on Information Systems* 35, 2, Article 15 (2016), 31 pages. doi:10.1145/2987380

[7] Shashank Gupta, Philipp Hager, Jin Huang, Ali Vardasbi, and Harrie Oosterhuis. 2024. Unbiased Learning to Rank: On Recent Advances and Practical Applications. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining* (Merida, Mexico) *(WSDM '24)*. Association for Computing Machinery, New York, NY, USA, 1118–1121. doi:10.1145/3616855.3636451

[8] Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Alberta, Canada) *(KDD '02)*. Association for Computing Machinery, New York, NY, USA, 133–142. doi:10.1145/775047.775067

[9] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in Web search. *ACM Trans. Inf. Syst.* 25, 2 (April 2007), 7–es. doi:10.1145/1229179.1229181

[10] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased Learning-to-Rank with Biased Feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* (Cambridge, United Kingdom) *(WSDM '17)*. Association for Computing Machinery, New York, NY, USA, 781–789. doi:10.1145/3018661.3018699

[11] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Foundations and Trends® in Information Retrieval* 3, 3 (2009), 225–331. doi:10.1561/1500000016

[12] Harry Markowitz. 1952. Portfolio Selection. *The Journal of Finance* 7, 1 (1952), 77–91. http://www.jstor.org/stable/2975974

[13] Harrie Oosterhuis. 2020. *Learning from User Interactions with Rankings: A Unification of the Field.* Ph. D. Dissertation. Informatics Institute, University of Amsterdam. https://harrieo.github.io//publication/2021-phd-thesis

[14] Harrie Oosterhuis. 2021. Computationally Efficient Optimization of Plackett-Luce Ranking Models for Relevance and Fairness. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 1023–1032. doi:10.1145/3404835.3462830

[15] Harrie Oosterhuis. 2022. Learning-to-Rank at the Speed of Sampling: Plackett-Luce Gradient Estimation with Minimal Computational Complexity. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2266–2271. doi:10.1145/3477495.3531842

[16] Harrie Oosterhuis. 2023. Doubly Robust Estimation for Correcting Position Bias in Click Feedback for Unbiased Learning to Rank. *ACM Trans. Inf. Syst.* 41, 3, Article 61 (Feb. 2023), 33 pages. doi:10.1145/3569453

[17] Harrie Oosterhuis, Rolf Jagerman, Zhen Qin, and Xuanhui Wang. 2026. Exposure-Based Reinforcement Learning to Rank. In *Proceedings of the 2026 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*.

[18] Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 Datasets. *CoRR* abs/1306.2597 (2013). http://arxiv.org/abs/1306.2597

[19] Tao Qin, Tie-Yan Liu, and Hang Li. 2010. A general approximation framework for direct optimization of information retrieval measures. *Information retrieval* 13, 4 (2010), 375–397.

[20] Oscar Rolando Ramirez Milian and Harrie Oosterhuis. 2026. An Epistemic Position-Based Click Model: From Interactions to Epistemic Distributions of Relevance and Bias. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Melbourne, VIC, Australia) *(SIGIR '26)*. Association for Computing Machinery, New York, NY, USA, 12 pages. doi:10.1145/3805712.3809529

[21] Tetsuya Sakai and Ruihua Song. 2011. Evaluating diversified search results using per-intent graded relevance. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Beijing, China) *(SIGIR '11)*. Association for Computing Machinery, New York, NY, USA, 1043–1052. doi:10.1145/2009916.2010055

[22] Pedro Henrique Silva Rodrigues, Daniel Xavier Sousa, Thierson Couto Rosa, and Marcos André Gonçalves. 2022. Risk-Sensitive Deep Neural Learning to Rank. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 803–813. doi:10.1145/3477495.3532056

[23] Ruihua Song, Zhenxiao Luo, Ji-Rong Wen, Yong Yu, and Hsiao-Wuen Hon. 2007. Identifying ambiguous queries in web search. In *Proceedings of the 16th International Conference on World Wide Web* (Banff, Alberta, Canada) *(WWW '07)*. Association for Computing Machinery, New York, NY, USA, 1169–1170. doi:10.1145/1242572.1242749

[24] Rikiya Takehi, Fernando Diaz, and Tetsuya Sakai. 2026. Diversification as Risk Minimization. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining* (USA) *(WSDM '26)*. Association for Computing Machinery, New York, NY, USA, 618–628. doi:10.1145/3773966.3777985

[25] Ali Vardasbi, Maarten de Rijke, and Ilya Markov. 2021. Mixture-Based Correction for Position and Trust Bias in Counterfactual Learning to Rank. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 1869–1878. doi:10.1145/3459637.3482275

[26] Jun Wang and Jianhan Zhu. 2009. Portfolio theory of information retrieval. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Boston, MA, USA) *(SIGIR '09)*. Association for Computing Machinery, New York, NY, USA, 115–122. doi:10.1145/1571941.1571963

[27] Lidan Wang, Paul N. Bennett, and Kevyn Collins-Thompson. 2012. Robust ranking models via risk-sensitive optimization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Portland, Oregon, USA) *(SIGIR '12)*. Association for Computing Machinery, New York, NY, USA, 761–770. doi:10.1145/2348283.2348385

[28] Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to Rank with Selection Bias in Personal Search. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pisa, Italy) *(SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 115–124. doi:10.1145/2911451.2911537

[29] Xuanhui Wang, Cheng Li, Nadav Golbandi, Michael Bendersky, and Marc Najork. 2018. The LambdaLoss Framework for Ranking Metric Optimization. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (Torino, Italy) *(CIKM '18)*. Association for Computing Machinery, New York, NY, USA, 1313–1322. doi:10.1145/3269206.3271784