Epistemic Click Models Through Evidential Deep Learning

Oscar Ramirez Milian
oscar.ramirezmilian@ru.nl

Harrie Oosterhuis
harrie.oosterhuis@ru.nl

1 Introduction

Click modelling and unbiased learning-to-rank methods estimate user preferences from logged clicks with rankings, while disentangling them from other causal factors that affect clicks [7, 10, 14, 25]. The position-based click model (PBM) [7, 8] is a foundational click model that remains relevant due to its combination of simplicity and effectiveness [1, 2, 6, 8, 14, 26, 28, 35]:

Definition 1.1 (*Position-based click model* (PBM)). *In the PBM, the probability of a click* ($\mathsf{C} = 1$) *on an item d displayed at position k of a ranking in the context q is a product of the probability that position k is examined* ($\mathsf{E} = 1$)*, conditioned on the position, and the probability that item is found attractive* ($\mathsf{R} = 1$)*, conditioned on the context and item:*

$$\mathbb{P}[\,\mathsf{C} = 1 \mid q, d, k\,] = \mathbb{P}[\,\mathsf{E} = 1 \mid k\,]\mathbb{P}[\,\mathsf{R} = 1 \mid q, d\,] = \theta_k \zeta_{q,d}. \quad (1)$$

The θ and ζ parameters are often referred to as *position bias* and *item relevance* (a.k.a. attractiveness or preference) factors respectively [8, 23]. Since examination and relevance cannot be observed directly, the PBM parameters have to be inferred from logged click data [7, 8, 35]. Let $N_{q,d,k} \in \mathbb{Z}_{\geq 0}$ be the number of times item d was displayed at position k and $M_{q,d,k} \in \mathbb{Z}_{\geq 0}$ the number of times d was clicked at k and let $\mathbf{N}$ and $\mathbf{M}$ be tuples of all $N_{q,d,k}$ and $M_{q,d,k}$ values respectively. Existing methods for estimating the parameters of the PBM take one of two approaches [25]; The click model family searches for the parameters that maximize the likelihood of the data [7, 15, 35]: $(\boldsymbol{\theta}^*, \boldsymbol{\zeta}^*) = \arg\max_{(\boldsymbol{\theta},\boldsymbol{\zeta})} \mathbb{P}[\,\mathbf{M} \mid \mathbf{N}, \boldsymbol{\theta}, \boldsymbol{\zeta}\,]$. Unbiased learning-to-rank methods first estimate the examination probabilities per rank, e.g., through position randomization [35], and then use these as propensities for inverse-propensity-scoring (IPS) estimates of relevance [14, 34], for example, with estimated propensities ρ: $\hat{\zeta}^{\text{IPS}}_{q,d} = \frac{1}{\sum_{k=1}^{K} N_{q,d,k}} \sum_{k=1}^{K} \frac{M_{q,d,k}}{\rho[\mathsf{E}=1|k]}$.

A significant limitation of these approaches is that they only provide pointwise estimates for the parameters [13]. Therefore, they do not give any indication of how reliable their estimates are, nor how much confidence one should have in their value [16]. This is especially concerning since the PBM explains observations by products of probabilities, and consequently, often many different parameter values can explain observed click data equally well [11, 25]. This larger limitation in the click modelling and unbiased learning-to-rank fields, as to the best of our knowledge, all their methodologies only provide pointwise estimates [10, 14, 15, 23, 27].

In this work, we propose the first epistemic parameter estimation method for the PBM, namely, our approach is based on evidential deep-learning [3, 21, 22, 31, 32] and learns a distribution for each of its parameters that represent epistemic uncertainty. Our model consists of a Beta distribution for each parameter, i.e., $\theta_k \in (0, 1)$ for every k and $\zeta_{q,d} \in (0, 1)$ for every (q, d), which are treated as independent samples. Accordingly, we optimize the model by searching for the best α and β parameters of the Beta distributions, i.e., (α_k, β_k) for every θ_k, and $(\alpha_{q,d}, \beta_{q,d})$ for every $\zeta_{q,d}$, by maximizing the likelihood of the data under the α and β parameters; with $\boldsymbol{\theta} \in (0, 1)^K$, $\boldsymbol{\zeta} \in (0, 1)^{QD}$, and their distribution parameters $\boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}, \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}$ as tuples of all α_k, β_k, $\alpha_{q,d}$ and $\beta_{q,d}$ respectively:

$$\underset{(\boldsymbol{\alpha_\theta},\boldsymbol{\beta_\theta},\boldsymbol{\alpha_\zeta},\boldsymbol{\beta_\zeta})}{\arg\max} \int \mathrm{P}\big(\mathbf{M} \mid \mathbf{N}, \boldsymbol{\zeta}, \boldsymbol{\theta}\big)\mathrm{p}\big(\boldsymbol{\zeta}, \boldsymbol{\theta} \,\big|\, \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}, \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}\big) d\boldsymbol{\zeta} d\boldsymbol{\theta}. \quad (2)$$

However, a direct application of Monte Carlo integration to solve (2) pose substantial challenges. In particular, the numerical precision of the sample values and the high variance inherent in the gradient estimation are the main difficulties. As a solution, we introduce a novel approach that optimizes a numerically-stable approximation of the log-likelihood and reduces variance through conditioning on partial samples.

We pose our novel method as the first *epistemic click model* and hope our work serves as a starting point for a Bayesian research direction in click modelling that further builds on evidential deep learning for uncertainty quantification [3, 17, 21, 29, 31].

2 Preliminaries: Notation and Epistemic PBM

In our setting, the observed data are query-item pairs, their displays to the users and the clicks received from displays. We have the query-item pairs $(q, d) \in \mathcal{D}$ with feature-vector representations $\mathbf{x}_{q,d}$. For every position $k \in \{1, 2, \ldots, K\}$ and pair (q, d), $N_{q,d,k} \in \mathbb{Z}_{\geq 0}$ indicates the number of times the item d was displayed at position k for query q, and $M_{q,d,k}$ the number of times it was clicked. For convenience, we also define $W_{q,d,k} = N_{q,d,k} - M_{q,d,k}$ as the number of displays without clicks. Furthermore, we define $\mathbf{N} = \big(N_{q,d,k} : (q, d) \in \mathcal{D}, k \in (1, 2, \ldots, K)\big)$ as the sequence of all $N_{q,d,k}$ values and define $\mathbf{M}$ and $\mathbf{W}$ analogously. Lastly, we assume that the number of pairs is much greater than the number of display positions: $|\mathcal{D}| >> K$, which is generally true in recommendation.

The PBM models the probability of a click as a product of a position factor θ_k and a relevance factor $\zeta_{q,d}$ (Definition 1.1). Furthermore, we assume that all clicks are independent Bernoulli variables, an assumption that often goes implicitly with the PBM [7]. Therefore, we can model each sum of clicks $M_{q,d,k} \sim \text{Bin}(N_{q,d,k}, \theta_k \zeta_{q,d})$ as binomial distributed random variable. To keep our notation brief, we define $\boldsymbol{\theta} = (\theta_k : k \in (1, 2, \ldots, K))$ as the sequence of all θ_k and define $\boldsymbol{\zeta}$ analogously. Thereby, $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$ capture the *aleatoric* uncertainty of our model [13], as they aim to describe the stochasticity inherent in the clicking behavior of users.

To capture *epistemic* uncertainty, we opt for an evidential approach [3, 19, 21] by fitting a parametric distribution over the possible values of $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$. Specifically, we rely on Beta distributions

CONSEQUENCES '25, Prague, Czech Republic

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

as each individual θ or ζ variable is in the closed interval $[0, 1]$. Accordingly, we model each θ_k as a sample from a Beta distribution specific to position k: $\theta_k \sim \text{Beta}(\alpha_k, \beta_k)$; and each $\zeta_{q,d}$ as a sample from a Beta distribution specific to the query-document pair q, d: $\zeta_{q,d} \sim \text{Beta}(\alpha_{q,d}, \beta_{q,d})$. We take the density function as the product of the individual marginal densities:

$$\mathrm{p}(\boldsymbol{\theta}, \boldsymbol{\zeta} \,|\, \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}, \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}) = \mathrm{p}(\boldsymbol{\theta} \,|\, \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta})\mathrm{p}(\boldsymbol{\zeta} | \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}). \quad (3)$$

This implies that θ_k or $\zeta_{q,d}$ samples are independent, i.e., conditioning on one does not change the distributions of the others:

$$\begin{aligned}\mathrm{p}(\boldsymbol{\theta} \,|\, \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}) &= \textstyle\prod_{k=1}^{K} \mathrm{p}(\theta_k | \alpha_k, \beta_k),\\ \mathrm{p}(\boldsymbol{\zeta} \,|\, \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}) &= \textstyle\prod_{q,d \in \mathcal{D}} \mathrm{p}(\zeta_{q,d} | \alpha_{q,d}, \beta_{q,d}).\end{aligned} \quad (4)$$

Again for brevity, we define $\boldsymbol{\alpha_\theta} = (\alpha_k : k \in (1, 2, \ldots, K))$, $\boldsymbol{\alpha_\zeta} = (\alpha_{q,d} : (q, d) \in \mathcal{D})$ and analogously $\boldsymbol{\beta_\theta}$ and $\boldsymbol{\beta_\zeta}$. Thus, $\boldsymbol{\alpha_\theta}$, $\boldsymbol{\beta_\theta}$, $\boldsymbol{\alpha_\zeta}$ and $\boldsymbol{\beta_\zeta}$ describe the *epistemic* uncertainty of our model, i.e., how probable each possible value of $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$ is predicted to be. To the best of our knowledge, this is the first epistemic PBM and the first *epistemic click model* altogether [7, 15, 18].

Finally, the question now becomes how to find the values for $\boldsymbol{\alpha_\theta}$, $\boldsymbol{\beta_\theta}$, $\boldsymbol{\alpha_\zeta}$ and $\boldsymbol{\beta_\zeta}$ such that our epistemic model best predicts the true $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$ values and an appropriate uncertainty about its predictions. Following evidential deep learning methodology [3, 20], we search for a model that maximizes the likelihood of the observed data:

$$\mathcal{L} = \mathbb{E}\Big[\prod_{q,d,k} \mathrm{P}\big(M_{q,d,k} \,\big|\, N_{q,d,k}, \theta_k, \zeta_{q,d}\big) \,\Big|\, \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}, \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta} \Big]. \quad (5)$$

Accordingly, our goal is to develop a method for optimizing $\boldsymbol{\alpha_\theta}$, $\boldsymbol{\beta_\theta}$, $\boldsymbol{\alpha_\zeta}$ and $\boldsymbol{\beta_\zeta}$ to maximize the likelihood of our observed data. Because the number of positions K is limited, we learn $\boldsymbol{\alpha_\theta}$ and $\boldsymbol{\beta_\theta}$ directly (in a lookup table). Conversely, since the number of document-query pairs is enormous, we learn a neural network to predict $\alpha_{q,d}$ and $\beta_{q,d}$ for a given feature vector $\mathbf{x}_{q,d}$, thereby making our approach generalizable to previously-unseen queries and documents.

3 Background: Naïve Monte Carlo Estimation

Since the likelihood is an expectation (5), it can be estimated unbiasedly through a straightforward Monte Carlo estimation [12]. We can take S samples for each of the θ and ζ parameters according to $\boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}$ and $\boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}$, which we group in vectors for convenience:

$$\tilde{\theta}_k^{(i)} \sim \text{Beta}(\alpha_k, \beta_k), \qquad \tilde{\boldsymbol{\theta}}^{(i)} = (\tilde{\theta}_1^{(i)}, \ldots, \tilde{\theta}_K^{(i)}), \quad (6)$$

$$\tilde{\zeta}_{q,d}^{(i)} \sim \text{Beta}(\alpha_{q,d}, \beta_{q,d}), \qquad \tilde{\boldsymbol{\zeta}}^{(i)} = (\tilde{\zeta}_{q,d}^{(i)} : (q, d) \in \mathcal{D}). \quad (7)$$

We can evaluate the probability $\tilde{L}^{(i)} = \tilde{L}(\tilde{\boldsymbol{\zeta}}^{(i)}, \tilde{\boldsymbol{\theta}}^{(i)})$ of our observations, conditioned on the value of a single sample of the position bias $\tilde{\boldsymbol{\theta}}^{(i)}$ and the relevance values $\tilde{\boldsymbol{\zeta}}^{(i)}$:

$$\tilde{L}^{(i)} \triangleq \prod_{q,d,k} \binom{N_{q,d,k}}{M_{q,d,k}} \Big(\tilde{\theta}_k^{(i)} \tilde{\zeta}_{q,d}^{(i)}\Big)^{M_{q,d,k}} \Big(1 - \tilde{\theta}_k^{(i)} \tilde{\zeta}_{q,d}^{(i)}\Big)^{W_{q,d,k}}. \quad (8)$$

The mean value of $\tilde{L}^{(i)}$ provides an unbiased estimate of the likelihood of our model, additionally, the gradient of the likelihood can be unbiasedly estimated with the log-derivative trick [36]:

$$\tilde{\mathcal{L}} \approx \frac{1}{S} \sum_{i=1}^{S} \tilde{L}^{(i)}, \quad \boldsymbol{\nabla}\mathcal{L} \approx \frac{1}{S} \sum_{i=1}^{S} \tilde{L}^{(i)} \Big(\boldsymbol{\nabla} \log \mathrm{p}(\tilde{\boldsymbol{\zeta}}^{(i)} | \boldsymbol{\alpha_\zeta}, \boldsymbol{\beta_\zeta}) + \boldsymbol{\nabla} \log \mathrm{p}(\tilde{\boldsymbol{\theta}}^{(i)} | \boldsymbol{\alpha_\theta}, \boldsymbol{\beta_\theta}) \Big). \quad (9)$$

4 Method: Optimizing the Epistemic PBM

The main challenge of estimating (9) is that it involves averaging over incredibly small values, this results in numerical precisions issues and exuberates variance-related problems during optimization. We propose *numerically stable computation, conditioning on partial samples* and *self-normalization* to overcome this challenge.

4.1 Numerical stability for loss estimation

Consider the log-value of the likelihood: $\ell(\boldsymbol{\theta}, \boldsymbol{\zeta}) \triangleq \log(L(\boldsymbol{\theta}, \boldsymbol{\zeta}))$. Accordingly, the mean in (9) can done in the log space with these log-values $\tilde{\ell}^{(i)}$ of the likelihood samples $\tilde{L}^{(i)}$:

$$\log(\tilde{\mathcal{L}}) = \log\Big(\frac{1}{S} \sum_{i=1}^{S} \mathrm{e}^{\tilde{\ell}^{(i)}} \Big) = \log\Big(\sum_{i=1}^{S} \mathrm{e}^{\tilde{\ell}^{(i)}} \Big) - \log(S). \quad (10)$$

The log-sum-exp operation [4], LSE : $\mathbb{R}^S \to \mathbb{R}$, is commonly used to avoid underflow when operating very small numbers. The LSE can thus give a more stable estimate of the log-likelihood:

$$\log(\tilde{\mathcal{L}}) = \text{LSE}(\tilde{\ell}^{(1)}, \ldots, \tilde{\ell}^{(S)}) - \log(S). \quad (11)$$

We note that while LSE improves stability, it is not immune to numerical precision errors. In particular, if one of the samples $\tilde{\ell}^{(i)}$ dominates the others by being much larger: $\forall j \neq i$, $\tilde{\ell}^{(i)} >> \tilde{\ell}^{(j)}$, then $\text{LSE}(\tilde{\ell}^{(1)}, \ldots, \tilde{\ell}^{(S)}) \approx \max(\tilde{\ell}^{(1)}, \ldots, \tilde{\ell}^{(S)}) = \tilde{\ell}^{(i)}$. Since: we can expect an underestimate of the log-likelihood. A key-insight is that this issue is less likely if the values over which the LSE is computed are closer to each other.

4.2 Conditioning on position bias

For brevity, we use $J_{q,d,k}$ to denote the probability of $M_{q,d,k}$ clicks given $N_{q,d,k}$ displays and specific values for θ_k and $\zeta_{q,d}$:

$$J_{q,d,k}(\theta_k, \zeta_{q,d}) = J_{q,d,k} = \mathrm{P}(M_{q,d,k} | N_{q,d,k}, \theta_k, \zeta_{q,d}). \quad (12)$$

Similarly, the probability of the observed clicks at every position for a single item-pair with comparable conditionals is:

$$J_{q,d}(\boldsymbol{\theta}, \zeta_{q,d}) = J_{q,d} = \prod_{k} J_{q,d,k}(\theta_k, \zeta_{q,d}). \quad (13)$$

We proceed by replacing the conditional on the specific values of the relevance $\zeta_{q,d}$ with our epistemic parameters $\alpha_{q,d}, \beta_{q,d}$, however, we keep the condition on specific values for the position bias $\boldsymbol{\theta}_k$:

$$\mathcal{J}_{q,d}(\boldsymbol{\theta}, \alpha_{q,d}, \beta_{q,d}) = \mathcal{J}_{q,d} = \mathop{\mathbb{E}}_{\zeta_{q,d}} \Big[J_{q,d}(\boldsymbol{\theta}, \zeta_{q,d}) \big| \boldsymbol{\theta} \Big]. \quad (14)$$

We rewrite the likelihood (5) as:

$$\mathcal{L} = \mathop{\mathbb{E}}_{\boldsymbol{\theta}} \Big[\prod_{q,d} \mathop{\mathbb{E}}_{\zeta_{q,d}} \Big[J_{q,d} \,\Big|\, \boldsymbol{\theta} \Big] \Big] = \mathop{\mathbb{E}}_{\boldsymbol{\theta}} \Big[\prod_{q,d} \mathcal{J}_{q,d} \Big]. \quad (15)$$

The variance of an expectation conditioned on a variable is guaranteed to be smaller or equal than the total variance.

To construct an estimator, we first divide our sampling procedure to separately take S_{pos} samples $\boldsymbol{\theta}^{(i)}$ and S_{rel} samples $\zeta_{q,d}^{(j)}$ for every query-item pair. Subsequently, we can estimate $\mathcal{J}_{q,d}$ using a single sample $\boldsymbol{\theta}^{(i)}$ while averaging over the S_{rel} samples $\zeta_{q,d}^{(j)}$:

$$\tilde{\mathcal{J}}_{q,d}\big(\boldsymbol{\theta}^{(i)}, \alpha_{q,d}, \beta_{q,d}\big) = \mathcal{J}_{q,d}^{(i)} = \frac{1}{S_{\text{rel}}} \sum_{j=1}^{S_{\text{rel}}} \tilde{J}_{q,d}\big(\theta_k^{(i)}, \zeta_{q,d}^{(j)}\big). \tag{16}$$

Based on (15), we estimate the log-likelihood with LSE:

$$\log\big(\tilde{\mathcal{L}}\big) \approx \text{LSE}\Big(\sum_{q,d} \log \tilde{\mathcal{J}}_{q,d}^{(1)}, \dots, \sum_{q,d} \log \tilde{\mathcal{J}}_{q,d}^{(S_{\text{pos}})}\Big) - \log\big(S_{\text{pos}}\big). \tag{17}$$

Importantly, the intermediate estimator (16) only concerns an expectation over a single random variable $\zeta_{q,d}$, and (given the intermediate estimates) the log-likelihood estimator (17) only concerns an expectation over the K random variables in $\boldsymbol{\theta}$. Thereby, the variance of both these estimators is lower than for (9).

4.3 Gradient estimation with self-normalization

Inspired by self-normalized importance sampling [5, 33], we apply self-normalization by dividing the $\tilde{L}^{(i)}$ values by the estimated overall likelihood $\tilde{\mathcal{L}}$; we note that this is equivalent to optimizing the log-likelihood since: $\boldsymbol{\nabla} \log(\mathcal{L}) = \frac{\boldsymbol{\nabla} \mathcal{L}}{\mathcal{L}}$. Importantly, this maintains the direction of the gradient whilst also normalizing the small values for $\tilde{L}^{(i)}$. Furthermore, self-normalization can be computed more stably with a softmax $\boldsymbol{\sigma}$ using LSE:

$$\boldsymbol{\sigma}_i(z_1, \cdots, z_S) \triangleq \boldsymbol{\sigma}(z_i) \triangleq \mathrm{e}^{z_i - \text{LSE}(z_1, \dots, z_S)}. \tag{18}$$

Since the outermost expectation in (17) is over position bias samples, the $\boldsymbol{\nabla}_{\text{pos}}$ gradient after self-normalization is quite straightforward:

$$\boldsymbol{\nabla}_{\text{pos}} \log\big(\tilde{\mathcal{L}}\big) \triangleq \frac{1}{S_{\text{pos}}} \sum_{i=1}^{S_{\text{pos}}} \Big(\boldsymbol{\sigma}\Big(\sum_{q,d} \log \tilde{\mathcal{J}}_{q,d}^{(i)}\Big) \boldsymbol{\nabla}_{\text{pos}} \log \mathrm{p}\big(\tilde{\boldsymbol{\theta}}^{(i)} \,|\, \boldsymbol{\alpha}_{\boldsymbol{\theta}}, \boldsymbol{\beta}_{\boldsymbol{\theta}}\big)\Big).$$

For $\boldsymbol{\nabla}_{\text{rel}}$, we estimate the gradient of (17) with:

$$\boldsymbol{\nabla}_{\text{rel}} \log\big(\tilde{\mathcal{L}}\big) \approx \frac{1}{S_{\text{pos}}} \sum_{i=1}^{S_{\text{pos}}} \left(\boldsymbol{\sigma}\Big(\sum_{q,d} \log \tilde{\mathcal{J}}_{q,d}^{(i)} \Big) \sum_{q,d} \frac{\boldsymbol{\nabla}_{\text{rel}} \tilde{\mathcal{J}}_{q,d}^{(i)}}{\tilde{\mathcal{J}}_{q,d}^{(i)}} \right), \tag{19}$$

where $\boldsymbol{\nabla}_{\text{rel}} \tilde{\mathcal{J}}_{q,d}^{(i)}$ is an estimate for the sample $\boldsymbol{\theta}^{(i)}$:

$$\boldsymbol{\nabla}_{\text{rel}} \tilde{\mathcal{J}}_{q,d}^{(i)} \approx \frac{1}{S_{\text{rel}}} \sum_{j=1}^{S_{\text{rel}}} \tilde{J}_{q,d}^{(i,j)} \boldsymbol{\nabla}_{\text{rel}} \log\Big(\mathrm{p}\big(\tilde{\zeta}_{q,d}^{(j)} \,|\, \alpha_{q,d}, \beta_{q,d}\big)\Big). \tag{20}$$

Thus, through numerically-stable self-normalization, we deal with near-zero likelihoods for samples; equivalently, self-normalization results in the gradients of the log-likelihood.

5 Experiments and Conclusion

We perform an experiment on two datasets: *MSLR-Web10K* [30] (Fold #1) and *Istella-S* [9] to evaluate how well our epistemic PBM can predict clicks and whether its epistemic distributions capture uncertainty appropriately. We simulate 10^5 query-interactions by uniformly sampling queries over the train, validation and test sets (with replacement). Simulated clicks are sampled from a PBM with $\theta_k = \frac{1}{k}$ and $\zeta_{q,d} = \frac{0.9}{4} y_{q,d} + 0.1$ where $y_{q,d} \in \{0, 1, 2, 3, 4\}$ are the datasets' relevance labels for item-query pairs. The logging policy ranker is a neural-network (2 hidden layers of 64 units) Plackett-Luce ranking model [24] optimized on 30 random training-set queries in a supervised manner (using the relevance labels). For each sampled query, a ranking with $K = 5$ positions is sampled from the policy ranker. In summary, to simulate click data, we sample 10^5 queries from each dataset, sample a ranking for each query sample, and subsequently, sample clicks for the ranking from the ground-truth PBM. For our epistemic PBM, we perform optimization 1500 epochs on a neural network (3 hidden layers of 64, 64 and 32 hidden units respectively) to predict the relevance parameters $\boldsymbol{\alpha}_{\zeta}$, $\boldsymbol{\beta}_{\zeta}$ and a lookup table for the K position bias parameters $\boldsymbol{\alpha}_{\boldsymbol{\theta}}$, $\boldsymbol{\beta}_{\boldsymbol{\theta}}$. All reported results are based on 15 independently repeated runs.

We start by evaluating *whether our proposed techniques increase the capability of epistemic PBM to predicting clicks.* Figure A1 display the learning curves in terms of log-likelihood of the test-set clicks over 1500 training epochs of models optimized with different combinations of our proposed techniques. Clearly, self-normalization and conditioning reach a much higher log-likelihood than the naïve estimator; baseline-corrections improve performance except when conditioning is also applied. The combination of self-normalization and conditioning reaches the highest performance and converges in the fewest epochs, therefore, we conclude that *our techniques improve the predictive abilities of our epistemic PBM considerably.*

Furthermore, we consider *whether the epistemic PBM can reach comparable predictive performance as the traditional pointwise PBM.* We turn to Figure A1 again and compare the performance of the pointwise PBM and the epistemic PBM optimized with self-normalization and conditioning. Surprisingly, the epistemic PBM outperforms the pointwise PBM immediately, and converges at a substantially higher log-likelihood. At first glance, this may be unexpected, as one may expect a tradeoff between the uncertainty quantification and likelihood maximization of the epistemic PBM, but no such tradeoff seems to be present. The pointwise PBM is known to be very sensitive to its initial parameters, i.e., Chuklin et al. [7] state that they should start near the true parameters for the best effectiveness, whereas we applied random initializations. In contrast, the epistemic PBM can start with a prior that gives all possible parameters equal epistemic probability, thereby avoiding tuning the initialization parameters, possibly explaining the difference. Regardless of the underlying reason, we conclude that *the epistemic PBM provides better predictive performance than the pointwise PBM.*

Finally, we wish to analyze *whether the learned distributions of the PBM represent epistemic uncertainty accurately.* Unfortunately, to the best of our knowledge, there exists no objective methodology for evaluating epistemic knowledge. Nevertheless, Figure A2 and A3 display several learned distributions for the position bias parameters, in addition to the pointwise predictions. From the placements of the means of the distributions, a similar trend as for the likelihood can be seen: the epistemic PBM with self-normalization and conditioning is more accurate at predicting the position bias parameters than the others. Intuitively, the widths of its distributions look appropriate; given that this model also has the highest likelihood, it appears to capture a true variability over possible parameter choices. Consequently, at best we can conclude that *our epistemic PBM with self-normalization and conditioning appears to be better at capturing prediction uncertainty than without those techniques and also better than the pointwise PBM.*

In this work, we introduced the first *epistemic click model* using evidential deep learning on epistemic distributions of a PBM.

Appendix

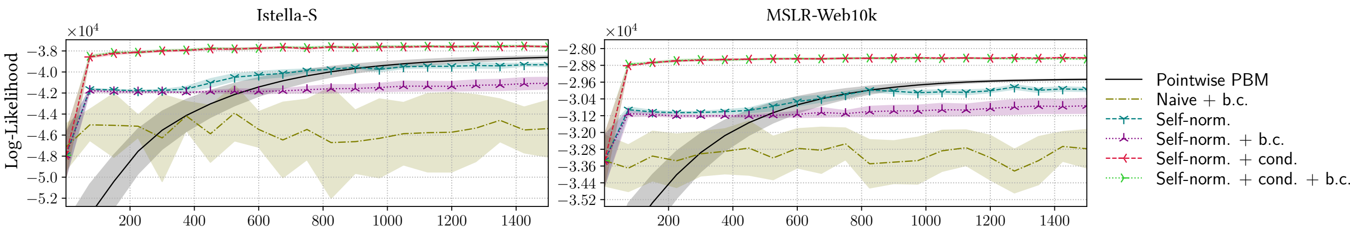

Figure A1: Log-Likelihood on test-set over training 1500 epochs as predicted by models trained with different methods: Naïve M.C. (9), baseline corrections (b.c.), conditioning (Section 4.2) and the traditional pointwise PBM and combinations of methods. X-axis: epochs; Y-axis: log-likelihood of clicks on test-set; The shaded areas estimate the 95% confidence intervals computed using a Student's t−distribution with 14 DoF, based on data from the 15 training sessions.

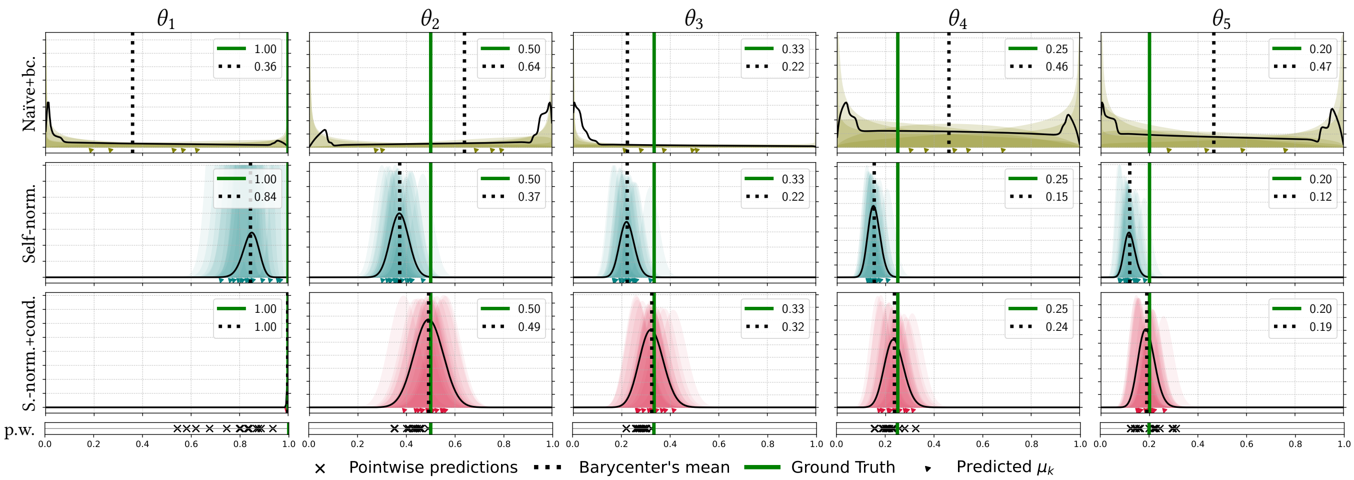

Figure A2: Learned epistemic position bias distributions and pointwise predictions (p.w.) on the Istella-S dataset. The X-axes represent a θ_k domain. The Y-axis the value of predicted distribution for θ_k by different methods.

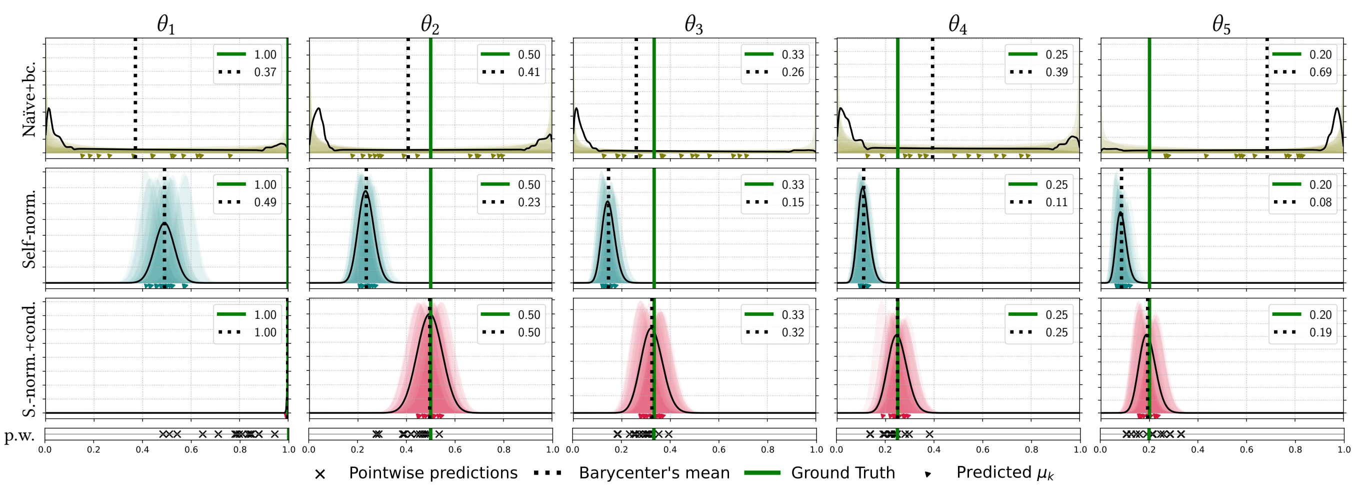

Figure A3: Learned epistemic position bias distributions and pointwise predictions (p.w.) on the MSLR-Web10k dataset. The X-axes represent a θ_k domain. The Y-axis the value of predicted distribution for θ_k by different methods.

References

[1] Qingyao Ai, Keping Bi, Cheng Luo, Jiafeng Guo, and W Bruce Croft. 2018. Unbiased learning to rank with unbiased propensity estimation. In *The 41st international ACM SIGIR conference on research & development in information retrieval*. 385–394.

[2] Qingyao Ai, Tao Yang, Huazheng Wang, and Jiaxin Mao. 2021. Unbiased learning to rank: online or offline? *ACM Transactions on Information Systems (TOIS)* 39, 2 (2021), 1–29.

[3] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. 2020. Deep evidential regression. *Advances in neural information processing systems* 33 (2020), 14927–14937.

[4] Pierre Blanchard, Desmond J Higham, and Nicholas J Higham. 2021. Accurately computing the log-sum-exp and softmax functions. *IMA J. Numer. Anal.* 41, 4 (2021), 2311–2330.

[5] Gabriel Cardoso, Sergey Samsonov, Achille Thin, Eric Moulines, and Jimmy Olsson. 2022. BR-SNIS: Bias Reduced Self-Normalized Importance Sampling. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 716–729. https://proceedings.neurips.cc/paper_files/paper/2022/file/04bd683d5428d91c5fbb5a7d2c27064d-Paper-Conference.pdf

[6] Ye Chen and Tak W Yan. 2012. Position-normalized click prediction in search advertising. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 795–803.

[7] Aleksandr Chuklin, Ilya Markov, and Maarten De Rijke. 2022. *Click models for web search*. Springer Nature.

[8] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the 2008 international conference on web search and data mining*. 87–94.

[9] 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. https://doi.org/10.1145/2987380

[10] 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. https://doi.org/10.1145/3616855.3636451

[11] Philipp Hager, Onno Zoeter, and Maarten de Rijke. 2025. Unidentified and Confounded? Understanding Two-Tower Models for Unbiased Learning to Rank. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)* (Padua, Italy) *(ICTIR '25)*. Association for Computing Machinery, New York, NY, USA, 347–357. https://doi.org/10.1145/3731120.3744604

[12] John Hammersley. 2013. *Monte Carlo methods*. Springer Science & Business Media.

[13] Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning* 110, 3 (2021), 457–506.

[14] 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*. 781–789.

[15] Jingwei Kang, Maarten de Rijke, Santiago de Leon-Martinez, and Harrie Oosterhuis. 2025. Rethinking Click Models in Light of Carousel Interfaces: Theory-Based Categorization and Design of Click Models. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)* (Padua, Italy) *(ICTIR '25)*. Association for Computing Machinery, New York, NY, USA, 44–55. https://doi.org/10.1145/3731120.3744585

[16] Armen Der Kiureghian and Ove Ditlevsen. 2009. Aleatory or epistemic? Does it matter? *Structural Safety* 31, 2 (2009), 105–112. https://doi.org/10.1016/j.strusafe.2008.06.020 Risk Acceptance and Risk Communication.

[17] Norman Knyazev and Harrie Oosterhuis. 2023. A Lightweight Method for Modeling Confidence in Recommendations with Learned Beta Distributions. In *Proceedings of the 17th ACM Conference on Recommender Systems* (Singapore, Singapore) *(RecSys '23)*. Association for Computing Machinery, New York, NY, USA, 306–317. https://doi.org/10.1145/3604915.3608788

[18] Jianping Liu, Yingfei Wang, Jian Wang, Meng Wang, and Xintao Chu. 2024. Probabilistic graph model and neural network perspective of click models for web search. *Knowledge and Information Systems* 66, 10 (2024), 5829–5873. https://doi.org/10.1007/s10115-024-02145-z

[19] Weiru Liu. 2001. *Propositional, Probabilistic and Evidential Reasoning: Integrating numerical and symbolic approaches*. Vol. 77. Springer Science & Business Media.

[20] Andrey Malinin, Sergey Chervontsev, Ivan Provilkov, and Mark Gales. 2020. Regression prior networks. *arXiv preprint arXiv:2006.11590* (2020).

[21] Andrey Malinin and Mark Gales. 2018. Predictive Uncertainty Estimation via Prior Networks. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2018/file/3ea2db50e62ceefceaf70a9d9a56a6f4-Paper.pdf

[22] Andrey Malinin and Mark Gales. 2019. Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness. *Advances in neural information processing systems* 32 (2019).

[23] 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

[24] 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. https://doi.org/10.1145/3404835.3462830

[25] Harrie Oosterhuis. 2022. Reaching the End of Unbiasedness: Uncovering Implicit Limitations of Click-Based Learning to Rank. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval* (Madrid, Spain) *(ICTIR '22)*. Association for Computing Machinery, New York, NY, USA, 264–274. https://doi.org/10.1145/3539813.3545137

[26] Harrie Oosterhuis. 2023. Doubly robust estimation for correcting position bias in click feedback for unbiased learning to rank. *ACM Transactions on Information Systems* 41, 3 (2023), 1–33.

[27] Harrie Oosterhuis, Rolf Jagerman, and Maarten de Rijke. 2020. Unbiased learning to rank: counterfactual and online approaches. In *Companion Proceedings of the Web Conference 2020*. 299–300.

[28] Zohreh Ovaisi, Ragib Ahsan, Yifan Zhang, Kathryn Vasilaky, and Elena Zheleva. 2020. Correcting for selection bias in learning-to-rank systems. In *Proceedings of The Web Conference 2020*. 1863–1873.

[29] Deep Shankar Pandey and Qi Yu. 2023. Evidential conditional neural processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 9389–9397.

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

[31] Murat Sensoy, Lance Kaplan, and Melih Kandemir. 2018. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems* 31 (2018).

[32] Ava P Soleimany, Alexander Amini, Samuel Goldman, Daniela Rus, Sangeeta N Bhatia, and Connor W Coley. 2021. Evidential deep learning for guided molecular property prediction and discovery. *ACS central science* 7, 8 (2021), 1356–1367.

[33] Adith Swaminathan and Thorsten Joachims. 2015. The Self-Normalized Estimator for Counterfactual Learning. In *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2015/file/39027dfad5138c9ca0c474d71db915c3-Paper.pdf

[34] 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*. ACM, 115–124.

[35] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018. Position bias estimation for unbiased learning to rank in personal search. In *Proceedings of the eleventh ACM international conference on web search and data mining*. 610–618.

[36] Ronald J Williams. 1992. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning* 8, 3-4 (1992), 229–256.