An Offline Metric for the Debiasedness of Click Models

Romain Deffayet[1,2] Philipp Hager[1,3] Jean-Michel Renders[2] Maarten de Rijke[1]

[1]University of Amsterdam [2]Naver Labs Europe [3]Mercury Machine Learning Lab

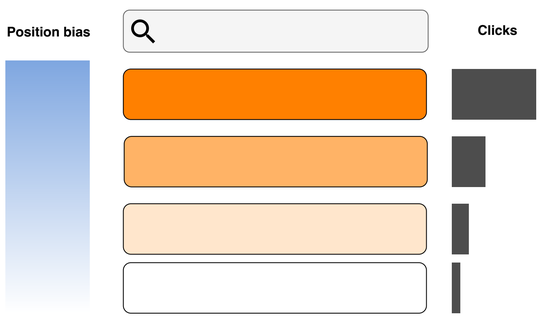
Click models

- How can we extract useful information about users from implicit feedback?
- **Click models explicitly model effects that impact clicks**: position, trust, item relevance, scrolling direction, session abandonment...
- **Applications**: understanding users, evaluation metrics, estimating biases, simulating users, and predicting ad clicks.

Most click model applications require out-of-distribution prediction, meaning predicting clicks on rankings not seen during training.

Evaluation fails to ensure that models generalize

- **Click prediction** using log-likelihood or perplexity only guarantees in-distribution performance [1].
- **Relevance assessment** using nDCG against expert annotations can fail when the system collecting the data is already good [1].



Ignoring position bias and naively interpreting clicks as relevance can achieve high nDCG scores

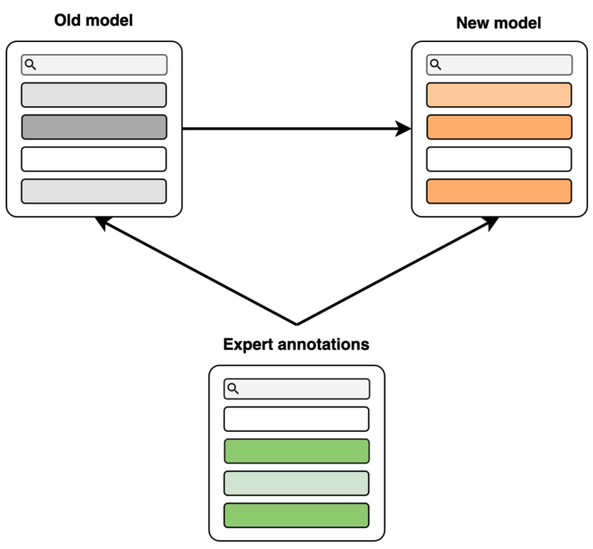
- **Blindly replicating the previous production system (without understanding users) can achieve high evaluation scores.**

Catch a model cheating

How would you detect a cheater in school?

- Comparing grades does not work, students who cheat can score high grades just by copying the answers of others.
- **We compare the mistakes students make!**

Using a small set of expert annotations, we can quantify if a new model makes similar mistakes to the previous model:



Debiasedness

Debiasedness: The predicted relevance of an item is not influenced by where the logging policy placed that item:

$$\tilde{R}^{\mathcal{D}} \perp R_l \mid (R, \mathcal{D})$$

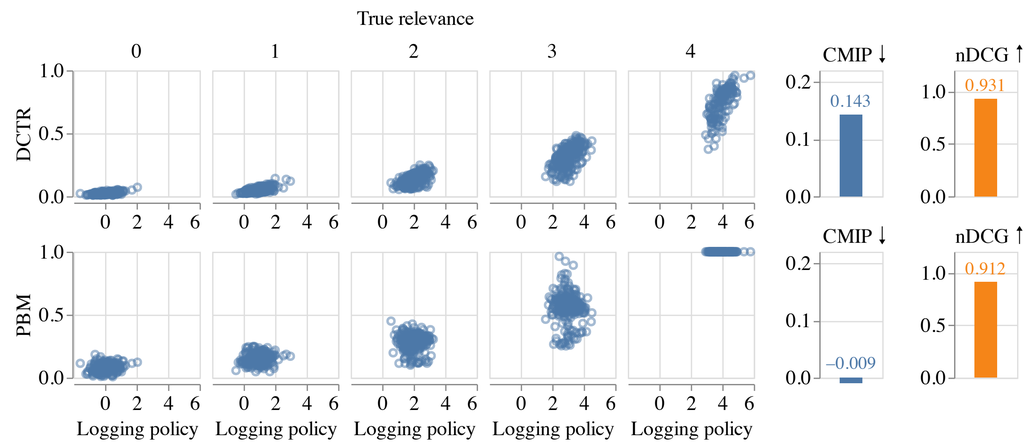
$\tilde{R}^{\mathcal{D}}$: Relevance estimations by a click model after training on dataset $\mathcal{D}$

R_l : Relevance estimated by the logging policy

R : Ground-truth relevance scores obtained by human annotators

We quantify the degree of debiasedness as the conditional mutual information w.r.t the logging policy (CMIP) [3, 2].

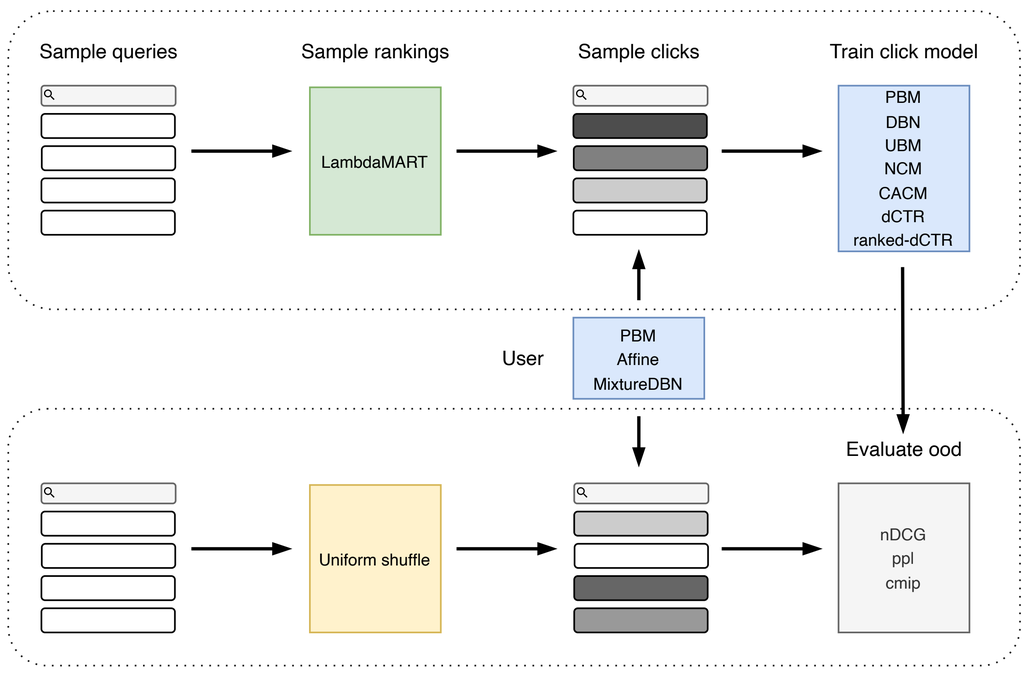
Visual intuition



A naive model (DCTR) outperforms an unbiased model (PBM) in terms of nDCG, but our CMIP metric catches that DCTR overfits on errors of the logging policy.

Simulating out-of-distributions settings

1. Rank items using one of three rankers (logging policies).
2. Sample train clicks on rankings using simulated users.
3. Train different neural click models.
4. Simulate test clicks on rankings obtained by a different policy (ood).
5. Measure click prediction performance of models on the ood test set.



Findings & Limitations

- **CMIP improves predicting the downstream performance of click models when coupled with existing metrics.**
- **CMIP helps to pick models that predict clicks well on unseen rankings.**

Limitations: CMIP is a pointwise metric, requires relevance annotations, and assumes that these annotations do not disagree with user preference.

References

[1] Deffayet, R., Renders, J.-M., and de Rijke, M. Evaluating the robustness of click models to policy distributional shift. In *ACM TOIS 2023*.

[2] Mukherjee, S., Asnani, H., and Kannan, S. CCMI: Classifier-based conditional mutual information estimation. In *UAI 2019*.

[3] Sen, R., Suresh, A. T., Shanmugam, K., Dimakis, A. G., and Shakkettai, S. Model-powered conditional independence test. In *NIPS 2017*.

University of Amsterdam

NAVER LABS Europe

Mercury machine learning lab

 r.e.deffayet@uva.nl, p.k.hager@uva.nl