Overview of the NTCIR-16 Unbiased Learning to Rank Evaluation (ULTRE) Task

Yurou Zhao, Zechun Niu, Feng Wang, Jiaxin Mao
Renmin University of China
P.R.C.
maojiaxin@gmail.com

Qingyao Ai, Tao Yang
University of Utah
USA
aiqy@cs.utah.edu

Junqi Zhang, Yiqun Liu
Tsinghua University
P.R.C.
yiqunliu@tsinghua.edu.cn

ABSTRACT

In this paper, we present an overview of the Unbiased Learning to Rank (ULTRE) task, a pilot task at the NTCIR-16. Motivated by the ongoing development of Unbiased Learning to Rank research, the ULTRE Task consists of two subtasks: offline ULTR and online ULTR. In this overview paper, we introduce the dataset, simulation method and evaluation protocols of ULTRE, and report the official evaluation results of the received runs.

KEYWORDS

Unbiased Learning to Rank, User Simulation, Web Search, Evaluation

1 INTRODUCTION

Unbiased learning to rank (ULTR) [2] that aims at learning a ranking model from the noisy and biased user clicks has become a trending topic in IR community. Existing ULTR research can be categorized into two groups: offline (counterfactual) LTR [7, 14] that learns an unbiased ranking model in an offline manner with batches of biased, historical click logs: 2) online ULTR [6, 9, 16] which makes online interventions of ranking and extracting unbiased feedback or deriving unbiased gradient for modeling training.

Despite the popularity of ULTR, how to properly evaluate and compare different ULTR approaches has not been systematically investigated. The widely-adopted, simulation-based evaluation method [1, 2] has several limitations: First, there are no standard click simulation settings or shared evaluation benchmarks for the ULTR community. Second, most studies only use a single user behavior model to simulate clicks, which may not fully capture the diverse patterns of real user behavior.

Therefore, due to increasing interests in ULTR and existing limitations in the previous evaluation method, we launch a pilot task-**Unbiased Learning to Ranking Evaluation (ULTRE)** [18] in NTCIR-16 to provide a shared benchmark and evaluation service for ULTR models specifically.

In ULTRE task, we extend and improve the click simulation phase in previous ULTR evaluation first. Then, we introduce the evaluation protocols and settings of two subtasks: **offline unbiased learning to rank** and **online unbiased learning to rank**, corresponding to two approaches of previous ULTR studies. The schedule of ULTRE is shown in Table 1.

The remainder of this paper is organized as follows: Section 2 details the dataset and click-simulation process of ULTRE. Section 3 introduces the evaluation protocols for two subtasks, respectively. Section 4 and Section 5 lists the submitted runs from the participants and reports the official results for each run. Finally, Section 6 gives a brief conclusion of this task.

Table 1: Schedule of ULTRE at NTCIR-16.

Time	Content
July 15, 2021	Dataset and simulated click logs released
August 15, 2021	Registration due
Sep 1, 2021 - Jan 15, 2022	Formal Run/Online evaluation
Feb 1, 2022	Final evaluation result released
Feb 1, 2022	Draft of task overview paper released
Mar 1, 2022	Participant paper submission due
May 1, 2022	Camera-ready paper submissions due
Jun 2022	NTCIR-16 Conference & EVIA 2022 in NII, Tokyo, Japan

Table 2: Statistics of ULTRE dataset.

	Training	Validation	Test
Unique queries	840	60	300
Sessions	111,911	60	300
Label	clicked or not (1) or (0)	relevance annotations (0-4)	relevance annotations (0-4)

2 DATASET AND CLICK SIMULATION

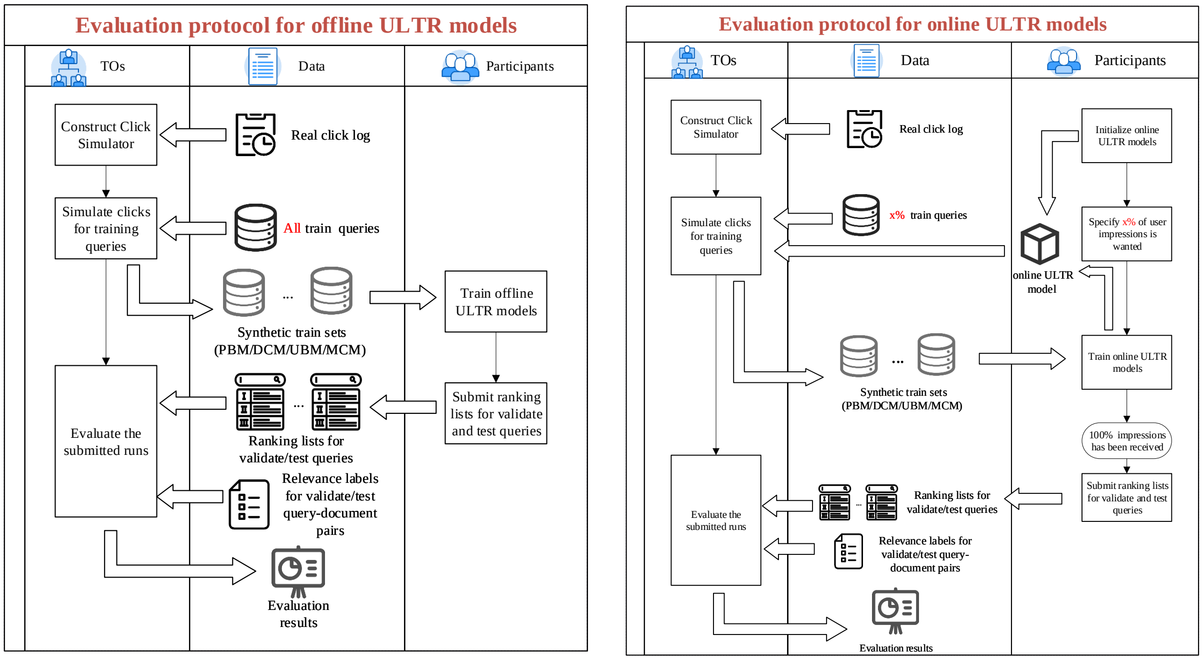
The ULTRE dataset is constructed based on SogouSRR[1] [17], a public dataset for relevance estimation and ranking in Web search. We select 1,200 unique queries from SogouSRR, 840 for training, 60 for validation and 300 for testing, and further collect the click logs and the HTML source codes of the landing pages for the top 10 search results of the queries.

Inspired by the widely-adopted simulation-based evaluation method, we first use the real click logs to train and calibrate the following click models[2]:

- Position-Based Model (PBM) [3]: a click model that assumes the click probability of a search result only depends on its relevance and its ranking position.

[1] http://www.thuir.cn/data-srr/

[2] For the detailed training procedure of these click models, please see [18]


Evaluation protocol for offline ULTR models
TOs
Data
Participants
Construct Click Simulator
Real click log
Simulate clicks for training queries
All train queries
...
Synthetic train sets (PBM/DCM/UBM/MCM)
Train offline ULTR models
Submit ranking lists for validate and test queries
Ranking lists for validate/test queries
Evaluate the submitted runs
Relevance labels for validate/test query-document pairs
Evaluation results
Evaluation protocol for online ULTR models
TOs
Data
Participants
Construct Click Simulator
Real click log
Initialize online ULTR models
Simulate clicks for training queries
x% train queries
Specify x% of user impressions is wanted
online ULTR model
Synthetic train sets (PBM/DCM/UBM/MCM)
Train online ULTR models
100% impressions has been received
Submit ranking lists for validate and test queries
Evaluate the submitted runs
Ranking lists for validate/test queries
Relevance labels for validate/test query-document pairs
Evaluation results

(a)

(b)

Figure 1: Evaluation protocols for offline and online ULTR models.

- Dependent Click Model (DCM) [5]: a click model that is based on the cascade assumption that the user will sequentially examine the results list and find attractive results to click until she feels satisfied with the clicked result.
- User Browsing Model (UBM) [4]: a click model that assumes the examination probability on a search result depends on its ranking position and the distance to the last clicked result.
- Mobile Click Model (MCM) [8]: a click model that considers the click necessity bias (i.e.some vertical results can satisfy users' information need without a click) in user clicks.

Equipped with the above click models, we then use 5 different simulation methods to generate the synthetic user clicks for 840 training queries as the training data for both the offline and online ULTR subtasks. We reuse the 4-level human relevance label set provided by SogouSRR to evaluate the ranking performance on the validation and test queries. Table 2 shows the details of the dataset we constructed.

3 EVALUATION PROTOCOLS

3.1 Evaluation protocol for Offline UTLR Subtask

Figure 1(a) shows the steps in the evaluation protocol of the offline subtask (TOs stands for the task organizers). The protocol consists three steps:

- Step 1: TOs generate simulated click logs by running various user simulation models on a "production" ranker for all the training queries.
- Step 2: Participants train their ULTR models on each synthetic train set respectively and submit the ranking lists (runs) for the test queries. Note that each run submitted by the participants is required to only use the synthetic data generated by a single click simulator, so ideally, for each ULTR model, we expect the participant to submit five runs.
- Step 3: TOs evaluate the runs based on true relevance labels in test set.

3.2 Evaluation protocol for Online UTLR Subtask

As shown in Figure 1(b), the evaluation protocol of online ULTR subtask involves similar steps in the offline one. However, the main difference between them is that the participants can iteratively submit the ranking lists to the TOs to get simulated clicks and use them to update their ULTR models in an online process.

- Step 1: Participants submit the ranking lists for training queries generated by their own ULTR models and specify how many user impressions they want to receive of the current model.
- Step 2: TOs sample the required number of queries from all training queries according to the query frequency in the

real log . Based on the ranking lists submitted by the participant in step 1, TOs generate simulated click logs by running the user simulation models for all the sampled queries and send them to the participant.

- `Step 3`: Participants update their models with the training data received in step 2.
- Repeat Step 1-Step 3 until participants receive approximately the same amount of training data used in the offline ULTR subtask.
- `Step N`: TOs evaluate on the submitted runs with the relevance labels for the test queries.

3.3 Statistical Significance Test

For each submitted run, we compute the nDCG@5 on the test set in Step 3 of the offline subtask and Step N of the online subtask, we further use the randomised Tukey HSD tests with B = 5, 000 trials [10] to test whether the nDCG@5 scores of each pair of runs are significantly different or not.

As we need a production ranker to produce ranking lists for training queries in the offline subtask, we trained a lambdaMART model [15] with 1% relevance labels randomly sampled from original training set (with 5-level relevance annotations). We also include this production ranker as baseline.

4 PARTICIPATION AND RECEIVED RUNS

Table 3 summarizes the statistics of the runs submitted by the participants. Although 9 teams have registered for the tasks, we only received 8 runs from 2 teams for offline ULTR subtask, and 7 runs from 2 teams for online ULTR subtask. The description for each submitted run will be included in the final version of the overview paper.

Table 3: ULTRE run statistics.

Team	offline	online
RUCIR21	5	4
UTIRL	3	3
total	8	7

4.1 Offline ULTR Subtask Runs

In offline ULTR subtask, RUCIR21 submitted 5 runs, namely SMCM, SMCMd, IPSMCMa, IPSMCM and DLAPBM. The SMCM utilizes mobile click model (MCM) [8]to estimate the relevance of training documents via EM algorithm, and uses the relevance estimation as a supervision signal to train LambdaMART. The SMCMd is similar to SMCM, except that it uses the relevance estimation to train deep neural networks (DNN). The IPSMCMa proposes a mobile click model-based inverse propensity score method and trains DNN. The IPSMCM is a simplified version of IPSMCMa. The DLAPBM revives the dual learning algorithm (DLA) [2], which jointly learns a DNN ranking model and a PBM propensity model.

In addition, UTIRL submitted 2 runs, namely PRS and ENSEMBLE. The PRS trains SetRank with the revived propensity ratio scoring (PRS) [13] algorithm. The ENSEMBLE is an ensemble of ten models consisting of five different algorithms on two neural networks, which averages the score of all the model for each document in the query.

4.2 Online ULTR Subtask Runs

In online ULTR subtask, RUCIR21 submitted 4 runs, namely PDGD, ODLA-PBM, ODLA-PBM2 and ODLA-MCM. The PDGD revives the pairwise differentiable gradient descent (PDGD) [9] algorithm and implements the ranking model with DNN. The ODLA-PBM is an online version of DLAPBM, which utilizes Plackett-Luce (PL) model to make online interventions. The ODLA-PBM2 is similar to ODLA-PBM, except that it uses all the click data collected so far to update the online DLA model. The ODLA-MCM tries to use MCM as an alternative to PBM and proposes an online DLA model based on MCM.

In addition, UTIRL submitted 3 runs, namely DBGD [16], MGD [11] and NSGD [12]. These three runs revive dueling bandit gradient descent (DBGD), multileave gradient descent (MGD), and null space gradient descent (NSGD) online algorithms respectively, and implements the ranking model with SetRank.

5 RESULTS

We evaluate each submitted run by computing nDCG@5 based on 5-level human relevance label for both offline and online subtask.

5.1 Offline ULTR Subtask Results

Table 4 and Figure 2(a) shows the nDCG@5 of each submitted runs for the offline subtask.

From the results, we can see that: 1) The PRS run of UTIRL team performed consistently well across different click simulation settings and achieved a highest averaged nDCG@5 score over the PBM, DCM, UBM, and MCM datasets. 2) The IPSMCMa run of RUCIR21 team achieved the highest nDCG@5 score on the FUSION dataset but its performance seemed to vary a lot on other datasets.

Table 6 shows the results of the statistical significance test for offline runs. In each subtable, we rank the runs in a descending order of nDCG@5. We use -/*/**/*** to indicate that the p-value $\geq 0.05/< 0.05/< 0.01/< 0.001$, respectively. If a run is significantly different (p-value ≤ 0.05) from another, we will also show the effect size (i.e. the standardized mean difference) in the Table 6.

From the results of the significant tests, we can observe that:

- There is no run that can perform significantly better than baseline when using PBM and MCM as the user simulation model.
- For DCM, UBM, and FUSION, only the best run is significantly better than the baseline.
- For FUSION, only the best run (RUCIR21-IPSMCMa) is significantly better than the baseline.
- Most runs are not significantly different from the others and most runs are not significantly better than the baseline.

5.2 Online ULTR Subtask Results

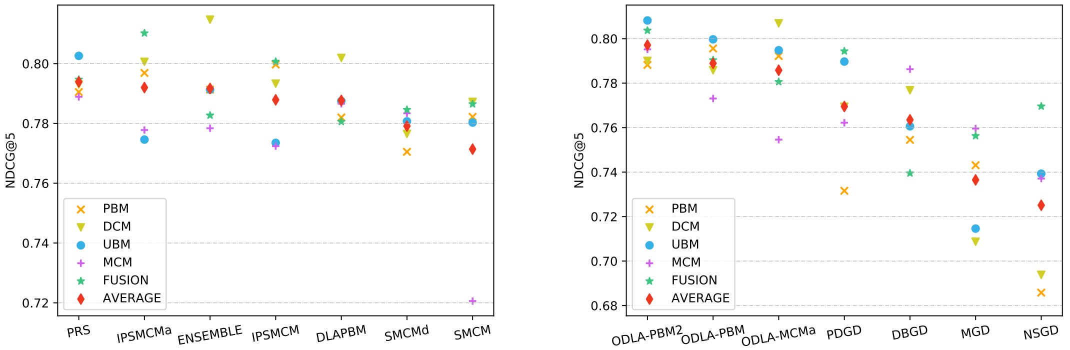
Similar to the offline result above, Table 5 and Figure 2(b) summarize the results in nDCG@5 for online runs.From the results, we can see that the ODLA-based runs (ODLA-PBM2, ODLA-PBM, ODLA-MCMa) seemed to perform consistently well in the online

Table 4: Official results of the Offline subtask. To highlight the best-performing runs under different user simulation models, we use bold numbers to indicate the best result and underlines to show the second-best result in each column.

No.	Team	Run	PBM	DCM	UBM	MCM	FUSION	AVG
0	-	baseline	0.7770					
1	UTIRL	PRS	0.7905	0.7930	**0.8026**	**0.7889**	0.7947	**0.7939**
2	RUCIR21	IPSMCMa	0.7969	0.8006	0.7746	0.7778	**0.8102**	0.7920
3	UTIRL	ENSEMBLE	0.7913	**0.8147**	0.7913	0.7784	0.7827	0.7917
4	RUCIR21	IPSMCM	**0.7997**	0.7933	0.7735	0.7724	0.8007	0.7879
5	RUCIR21	DLAPBM	0.7820	0.8019	0.7875	0.7866	0.7806	0.7877
6	RUCIR21	SMCMd	0.7705	0.7765	0.7807	0.7834	0.7846	0.7791
7	RUCIR21	SMCM	0.7822	0.7872	0.7803	0.7206	0.7865	0.7714

Table 5: Official results of the online subtask. To highlight the best-performing runs under different user simulation models, we use bold numbers to indicate the best result and underlines to show the second-best result in each column.

No.	Team	Run	PBM	DCM	UBM	MCM	FUSION	AVG
0	-	baseline	0.7770					
1	RUCIR21	ODLA-PBM2	0.7882	0.7900	**0.8082**	**0.7952**	**0.8037**	**0.7971**
2	RUCIR21	ODLA-PBM	**0.7956**	0.7858	0.7997	0.7731	0.7904	0.7889
3	RUCIR21	ODLA-MCMa	0.7922	**0.8069**	0.7948	0.7546	0.7806	0.7858
4	RUCIR21	PDGD	0.7316	0.7694	0.7897	0.7622	0.7944	0.7695
5	UTIRL	DBGD	0.7545	0.7768	0.7606	0.7863	0.7395	0.7635
6	UTIRL	MGD	0.7431	0.7087	0.7146	0.7596	0.7563	0.7365
7	UTIRL	NSGD	0.6858	0.6938	0.7393	0.7371	0.7696	0.7251



(a)

(b)

Figure 2: Official results of the Offline and Online subtask. 2(a): offline, 2(b): online.

subtask, especially when compared with the dueling-bandit-based runs. In addition, the ODLA-PBM2 run achieved better performance than the offline DLAPBM run. However, in general, the best runs in online subtask achieve similar performance to those best runs in offline subtask. Note that the selection bias occurs equally in both the offline and online evaluation protocols, which may diminish the advantage of online ULTR methods over offline ones.

Table 7 shows the results of the statistical significance test for the online runs.

From the significance test results, we can observe that:

- There is no run that perform significantly better than the baseline for PBM, DCM, and MCM simulation methods.
- For UBM and FUSION, only ODLA-PBM2 from RUCIR21 is significantly better than the baseline.
- For FUSION, only the best run (RUCIR21-ODLA-PBM2) is significantly better than the baseline. And only the worst run (UTIRL-DBGD) is significantly worse than the baseline.

Table 6: The results of the randomized Tukey HSD tests of the offline runs. -/*/ to indicate that the p-value $\geq 0.05/< 0.05/< 0.01$, respectively. For the significant pair, we show the effective size ES_{E2} in the table.**

a The results for PBM.

ES_{E2}	DLAPBM	IPSMCMa	IPSMCM	PRS	SMCM	baselnie	SMCMd
IPSMCM	-	-	-	-	-	-	$.31^{**}$
IPSMCMa		-	-	-	-	-	$.28^{*}$
ENSEMBLE			-	-	-	-	-
PRS				-	-	-	-
SMCM					-	-	-
DLAPBM						-	-
baseline							-

b The results for DCM.

ES_{E2}	DLAPBM	IPSMCMa	IPSMCM	PRS	SMCM	baseline	SMCMd
ENSEMBLE	-	-	-	-	$.27^{*}$	$.37^{**}$	$.37^{**}$
DLAPBM		-	-	-	-	-	-
IPSMCMa			-	-	-	-	-
IPSMCM				-	-	-	-
PRS					-	-	-
SMCM						-	-
baseline							-

c The results for UBM.

ES_{E2}	ENSEMBLE	DLAPBM	SMCMd	SMCM	baseline	IPSMCMa	IPSMCM
PRS	-	-	-	-	$.26^{*}$	$.29^{**}$	$.30^{**}$
ENSEMBLE		-	-	-	-	-	-
DLAPBM			-	-	-	-	-
SMCMd				-	-	-	-
SMCM					-	-	-
baseline						-	-
IPSMCMa							-

d The results for MCM.

ES_{E2}	DLAPBM	SMCMd	ENSEMBLE	IPSMCMa	baseline	IPSMCM	SMCM
PRS	-	-	-	-	-	-	$.63^{**}$
DLAPBM		-	-	-	-	-	$.61^{**}$
SMCMd			-	-	-	-	$.58^{**}$
ENSEMBLE				-	-	-	$.54^{**}$
IPSMCMa					-	-	$.53^{**}$
baseline						-	$.52^{**}$
IPSMCM							$.48^{**}$

e The results for FUSION.

ES_{E2}	IPSMCM	PRS	SMCMd	SMCM	ENSEMBLE	DLAPBM	baseline
IPSMCMa	-	-	$.26^{*}$	-	$.28^{*}$	$.30^{**}$	$.33^{**}$
IPSMCM		-	-	-	-	-	-
PRS			-	-	-	-	-
SMCMd				-	-	-	-
SMCM					-	-	-
ENSEMBLE						-	-
DLAPBM							-

- Most runs are not significantly different from others and most runs are not significantly better than the baseline.

6 CONCLUSIONS

This overview summarizes the dataset, simulation and evaluation methodology of NTCIR-16 ULTRE task and reports the official results of this task. In offline ULTR subtask, the PRS run from UTIRL team performs consistently well across different click simulation settings and the IPSMCMa run from RUCIR21 team achieves the highest nDCG@5 score on the FUSION dataset. In online ULTR subtask, the ODLA-based runs seem to perform consistently better than the bandit-based runs. However, both in the offline and online subtask, most of the differences between the submitted runs are not statistically significant, and only a few runs are significantly better than the baseline run. In addition, we find that evaluating the ULTR algorithms based on different click simulation methods yield varying results. In the future, we would like to further investigate the reliability and validity of the simulation-based evaluation.

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, Jiaxin Mao, Yiqun Liu, and W Bruce Croft. 2018. Unbiased learning to rank: Theory and practice. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management.* 2305–2306.

[3] 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.

[4] Georges E Dupret and Benjamin Piwowarski. 2008. A user browsing model to predict search engine click data from past observations.. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in*

information retrieval. 331–338.

[5] Fan Guo, Chao Liu, and Yi Min Wang. 2009. Efficient multiple-click models in web search. In *Proceedings of the second acm international conference on web search and data mining*. 124–131.

[6] Yiling Jia, Huazheng Wang, Stephen Guo, and Hongning Wang. 2021. Pairrank: Online pairwise learning to rank by divide-and-conquer. In *Proceedings of the Web Conference 2021*. 146–157.

[7] 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.

[8] Jiaxin Mao, Cheng Luo, Min Zhang, and Shaoping Ma. 2018. Constructing click models for mobile search. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 775–784.

[9] Harrie Oosterhuis and Maarten de Rijke. 2018. Differentiable unbiased online learning to rank. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1293–1302.

[10] Tetsuya Sakai. 2013. Metrics, statistics, tests. In *PROMISE winter school*. Springer, 116–163.

[11] Anne Schuth, Harrie Oosterhuis, Shimon Whiteson, and Maarten de Rijke. 2016. Multileave gradient descent for fast online learning to rank. In *Proceedings of the 9th ACM International Conference on Web Search and Data Mining*. 457–466.

[12] Huazheng Wang, Ramsey Langley, Sonwoo Kim, Eric McCord-Snook, and Hongning Wang. 2018. Efficient exploration of gradient space for online learning to rank. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 145–154.

[13] Nan Wang, Zhen Qin, Xuanhui Wang, and Hongning Wang. 2021. Non-clicks mean irrelevant? propensity ratio scoring as a correction. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 481–489.

[14] 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*. 115–124.

[15] Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. 2010. Adapting boosting for information retrieval measures. *Information Retrieval* 13, 3 (2010), 254–270.

[16] Yisong Yue and Thorsten Joachims. 2009. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning*. 1201–1208.

[17] Junqi Zhang, Yiqun Liu, Shaoping Ma, and Qi Tian. 2018. Relevance estimation with multiple information sources on search engine result pages. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 627–636.

[18] Yurou Zhao, Jiaxin Mao, and Qingyao Ai. 2021. ULTRE framework: a framework for Unbiased Learning to Rank Evaluation based on simulation of user behavior. (2021).

Table 7: The results of the randomized Tukey HSD tests of the online runs. -/*/ to indicate that the p-value $\geq 0.05/< 0.05/< 0.01$, respectively. For the significant pair, we show the effective size ES_{E2} in the table.**

a The results for PBM.

ES_{E2}	ODLA-MCMa	ODLA-PBM2	baseline	DBGD	MGD	PDGD	NSGD
ODLA-PBM	-	-	-	.33**	.43**	.52**	.89**
ODLA-MCMa		-	-	.31**	.40**	.49**	.86**
ODLA-PBM2			-	.27*	.37**	.46**	.83**
baseline				-	.28*	.37**	.74**
DBGD					-	-	.56**
MGD						-	.46**
PDGD							.37**

b The results for DCM.

ES_{E2}	PDGD	ODLA-PBM2	ODLA-PBM	MGD	baseline	DBGD	NSGD
ODLA-MCMa	.31**	-	-	.81**	-	-	.93**
PDGD		-	-	.50**	-	-	.62**
ODLA-PBM2			-	.67**	-	-	.79**
ODLA-PBM				.64**	-	-	.76**
MGD					.56**	.56**	-
baseline						-	.69**
DBGD							.69**

c The results for UBM.

ES_{E2}	ODLA-PBM	ODLA-MCMa	PDGD	baseline	DBGD	NSGD	MGD
ODLA-PBM2	-	-	-	.28*	.42**	.61**	.83**
ODLA-PBM		-	-	-	.35**	.54**	.76**
ODLA-MCMa			-	-	.30**	.49**	.71**
PDGD				-	-	.45**	.67**
baseline					-	.34**	.56**
DBGD						-	.41**
NSGD							-

d The results for MCM.

ES_{E2}	DBGD	baseline	ODLA-PBM	PDGD	MGD	ODLA-MCMa	NSGD
ODLA-PBM2	-	-	-	.31**	.33**	.38**	.54**
DBGD		-	-	-	-	.30**	.46**
baseline			-	-	-	-	.37**
ODLA-PBM				-	-	-	.34**
PDGD					-	-	-
MGD						-	-
ODLA-MCMa							-

e The results for FUSION.

ES_{E2}	PDGD	ODLA-PBM	ODLA-MCMa	baseline	NSGD	MGD	DBGD
ODLA-PBM2	-	-	-	.25*	.32**	.45**	.61**
PDGD		-	-	-	-	.36**	.52**
ODLA-PBM			-	-	-	.32**	.48**
ODLA-MCMa				-	-	-	.39**
baseline					-	-	.36**
NSGD						-	.29*
MGD							-