CIR at the NTCIR-17 ULTRE-2 Task

Lulu Yu, Keping Bi, Jiafeng Guo, Xueqi Cheng

CAS Key Lab of Network Data Science and Technology, ICT, CAS, Beijing, China

INTRODUCTION

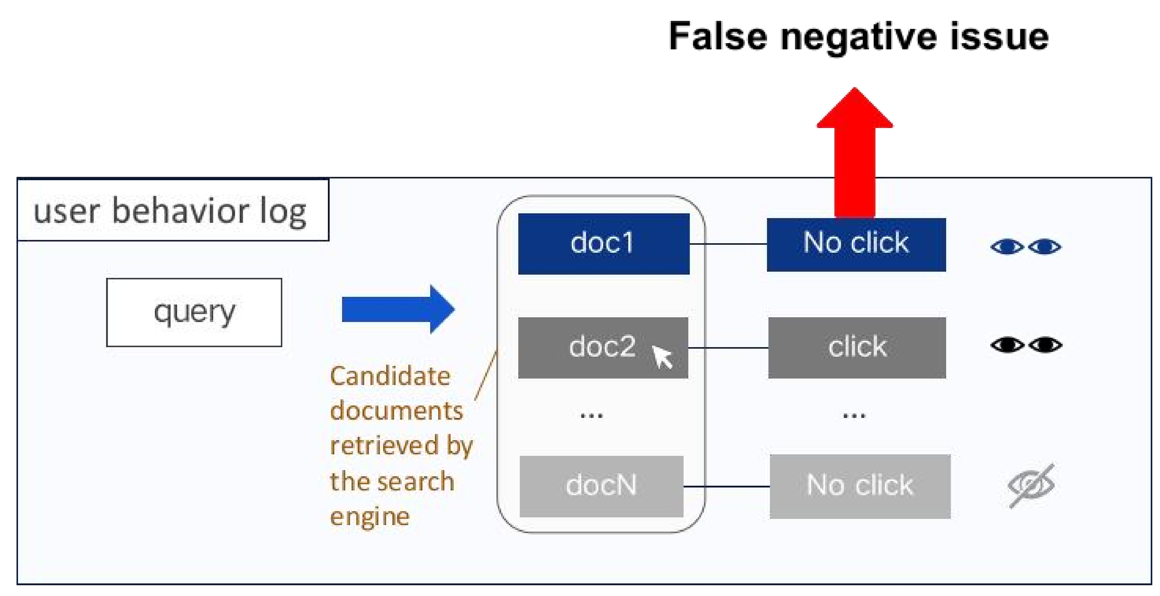
- Unbiased learning to rank (ULTR) aims to train an unbiased ranking model with biased user behavior logs.
- Although many ULTR models have achieved promising results on synthetic data, they still **lack guarantees of effectiveness in real-world scenarios**.
- In NTCIR-17, the ULTRE-2 task will evaluate the effectiveness of ULTR models with **a new, large-scale user behavior log collected from a commercial Web search engine Baidu**.

MOTIVATION

- The issue of false negatives is very severe in the Baidu search data, much more severe than the position bias.

<=> **Non-clicks do not mean irrelevant** due to the high quality of the search results returned by the search engine.

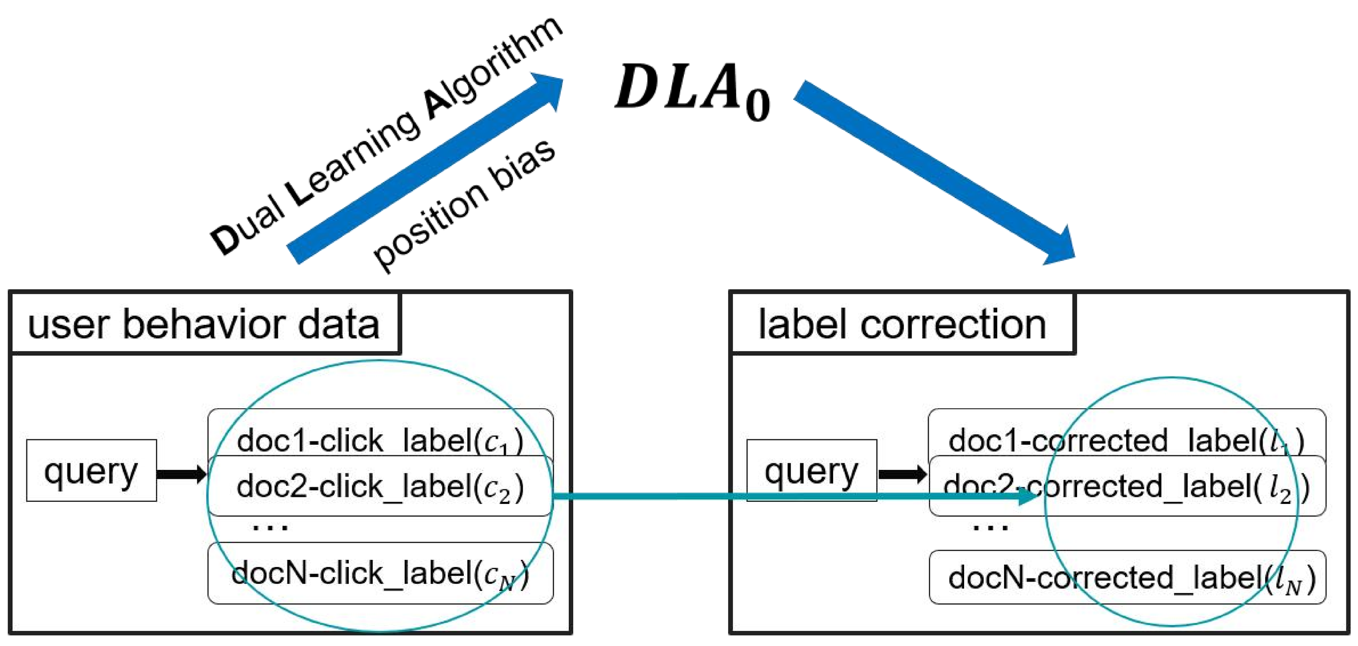
- Adopt the **Dual Learning Algorithm (DLA) to address the position bias** and use it **as an auxiliary model to study how to alleviate the false negative issue**.



OUR APPROACH

We approach the false negative issue **from two perspectives**.

- **Label Correction**
  - Correct the labels for non-clicked items by a relevance judgment model trained from DLA, and learn a new ranker that is initialized from DLA.

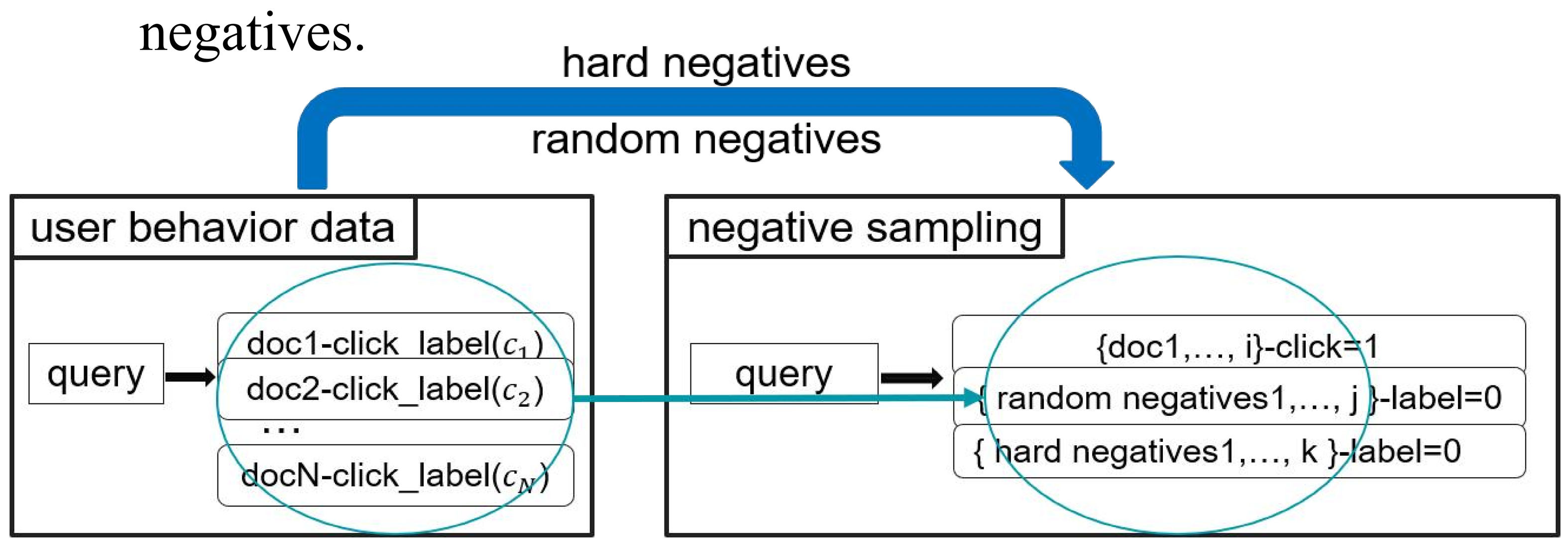


  - We attempt various reasonable strategies to transform the output of this auxiliary model into new labels.

$$l_{ij} = \begin{cases} sig: sigmoid(a_{ij}) \\ min: 1 \ if \ a_{ij} \geq min(a_{ik}),\ c_{ik} = 1 \end{cases}$$

  - where l_{ij}, a_{ij} means the label, DLA_0 output of the i^{th} query's j^{th} item with $c_{ij} = 0$.

- **Negative Sampling**
  - We try to rebuild the ranking lists to reduce the number of false negatives by adding random and hard negatives and replacing original non-click candidate items with random negatives.



  - We have devised two schemes:
    - "click-only" scheme replaces all non-clicked items with random negatives.
    - "last-click" scheme replaces items after the last clicked item with random negatives.
  - The loss function we use is as follows.

$$L_{click-only}/L_{last-click} = -\sum_{x \in \pi'_q} \frac{e^{f(x)}}{\sum_{x \in \pi'_q} e^{f(x)}}$$

EXPERIMENTS

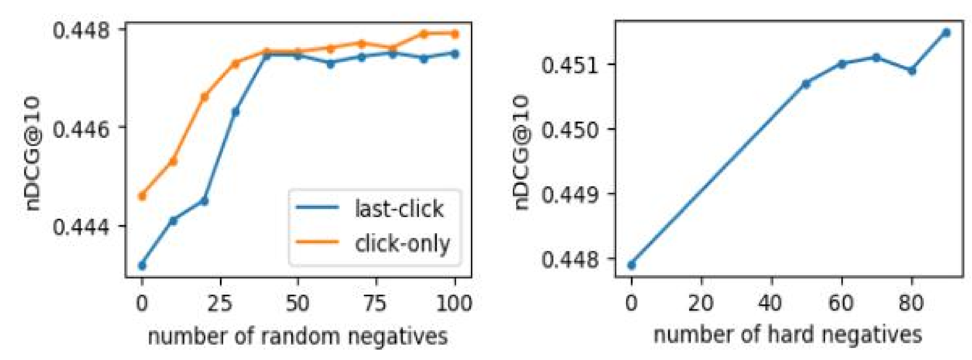
Effect of Label Correction

- DLA with Label Correction outperforms the basic DLA model, under various strategies.
  - The underline denotes the performance of the baseline DLA.

Model	nDCG@10 test	nDCG@10 total valid	nDCG@10 20% valid	DCG@10 test
Scratch-DLA-LC (sig)	**0.5355**	**0.5019**	/	11.4538
Aux-DLA-LC (sig)	0.5326	0.5015	/	11.3898
Scratch-DLA-LC (min)	unk	0.4816	/	unk
Aux-DLA-LC (min)	unk	0.4947	/	unk
<u>DLA</u>	<u>0.5247</u>	<u>0.4920</u>	/	<u>11.2031</u>
lgbBase	0.5350	/	0.5003	11.4794
lgbBaseAdd	0.5333	/	0.5021	11.4616

Effect of Negative Sampling

- We investigate the use of negative sampling on the validation set.
- The nDCG@10 on the validation set indicates that this approach is effective in improving performance.



Performance curves of two schemes ("click-only" and "last-click") w.r.t. the number of random and hard negatives. (a) Performance curves of two schemes w.r.t. the number of random negatives. (b) The Performance curve of the "click-only" scheme w.r.t. the number of hard negatives.

CONCLUSION

Motivation: Due to the high quality of the search results returned by the search engine, there exists severe false negative issue. Thus we propose two approaches to tackle this issue: label correction and negative sampling.

Method: DLA with Label Correction, Naïve Algorithm with Negative Sampling.

Effectiveness: Both methods can enhance the model performance and our best method has achieved nDCG@10 of 0.5355, which is 2.66% better than the best score from the organizer.