sigir21

Clicks can be Cheating: Counterfactual Recommendation for Mitigating Clickbait Issue

Wenjie Wang
wenjiewang96@gmail.com
National University of Singapore

Fuli Feng*
fulifeng93@gmail.com
Sea-NExT Joint Lab, Singapore
National University of Singapore

Xiangnan He
hexn@ustc.edu.cn
University of Science and Technology of China

Hanwang Zhang
hanwangzhang@ntu.edu.sg
Nanyang Technological University

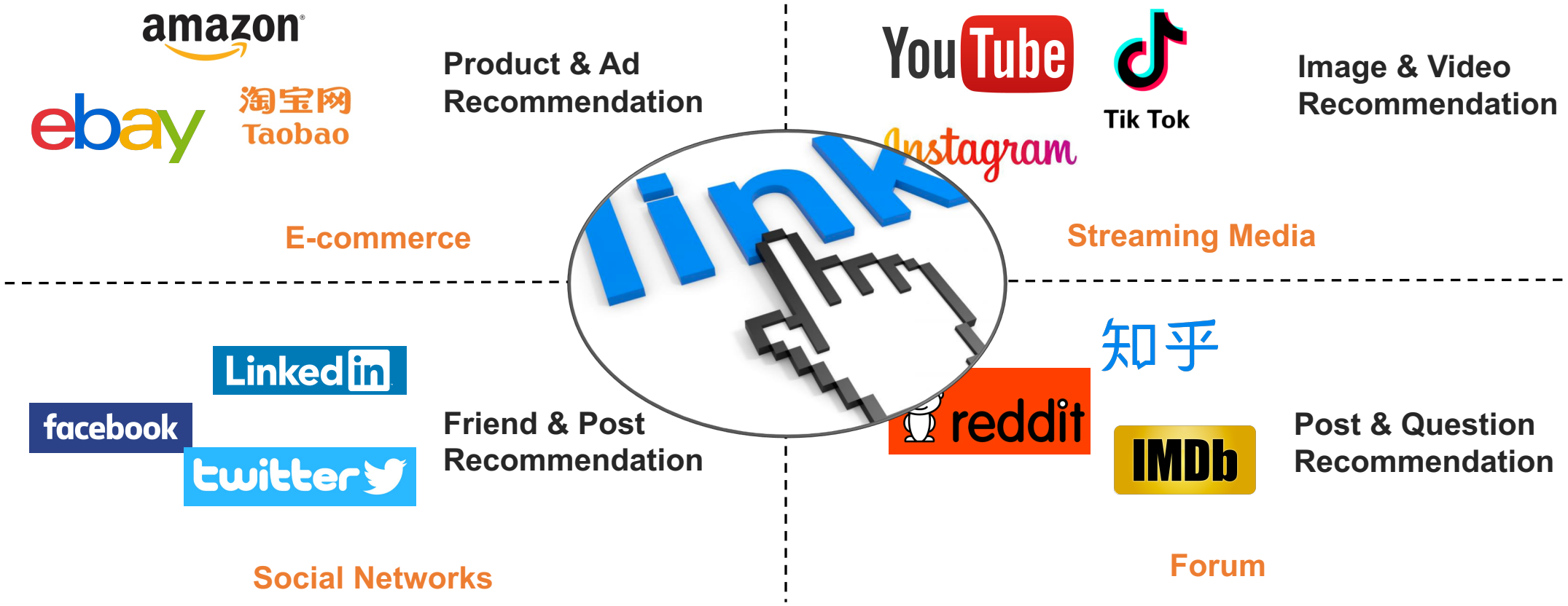
Tat-Seng Chua
dcscts@nus.edu.sg
National University of Singapore

Presenter: Wenjie WANG

Jun 2021

Background

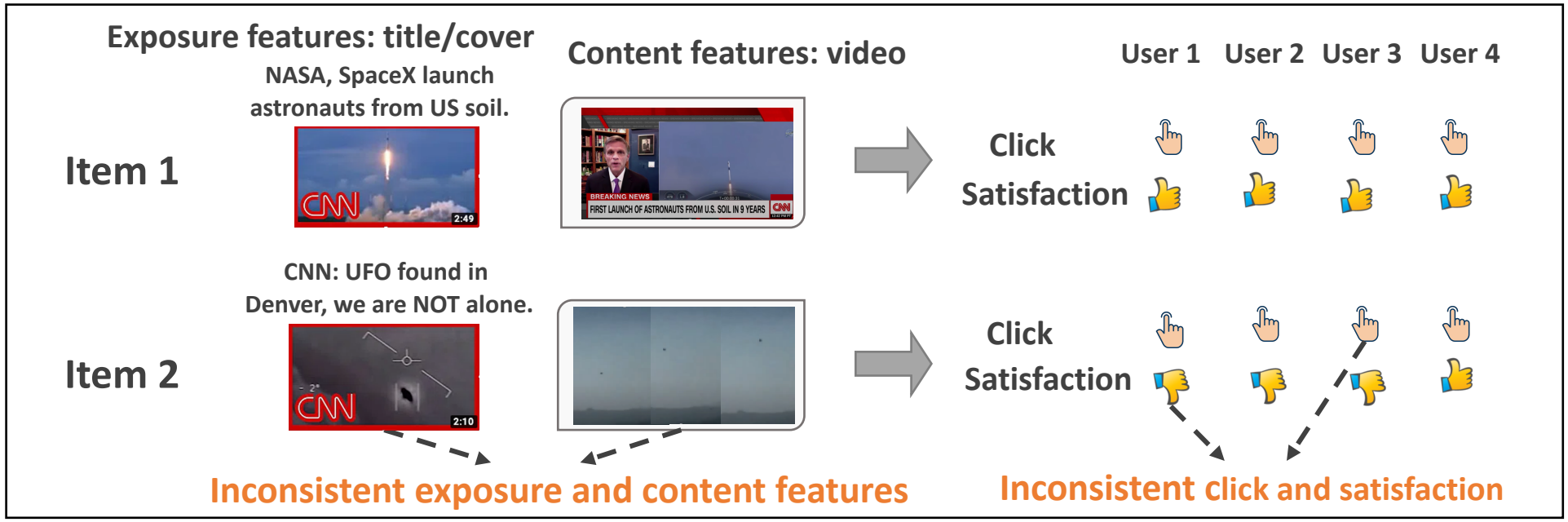
- Recommendation has been widely applied in online services.
- **Clicks** are popular to indicate user preference.



Background

Clickbait Issue

- ✓ **Clicks have intrinsic bias and various issues.**
- ✓ Users tend to click the items with attractive exposure features.
  - Also called attractiveness bias or caption bias.
  - **Exposure features**: available features **before** clicking, e.g., headline and cover image.
  - **Content features**: available features **after** clicking, e.g., video.
  - **Clickbait content**: deceptive or misleading exposure features.



Background

Clickbait Issue

- It is common that a user is “misled” to click an item by the attractive title/cover.
- Clickbait issue in recommendation: consequently, recommender model will recommend items with attractive exposure features but disappointing content features frequently.
- Negative effect of clickbait issue:
  - It is **unfair** to the items with high-quality video content.
  - The unfairness severely **hurts user’s trust and satisfaction** on the recommender system.
- Exposure features (*e.g.,* title/cover) attract users while content features (*e.g.,* video) are disappointing.

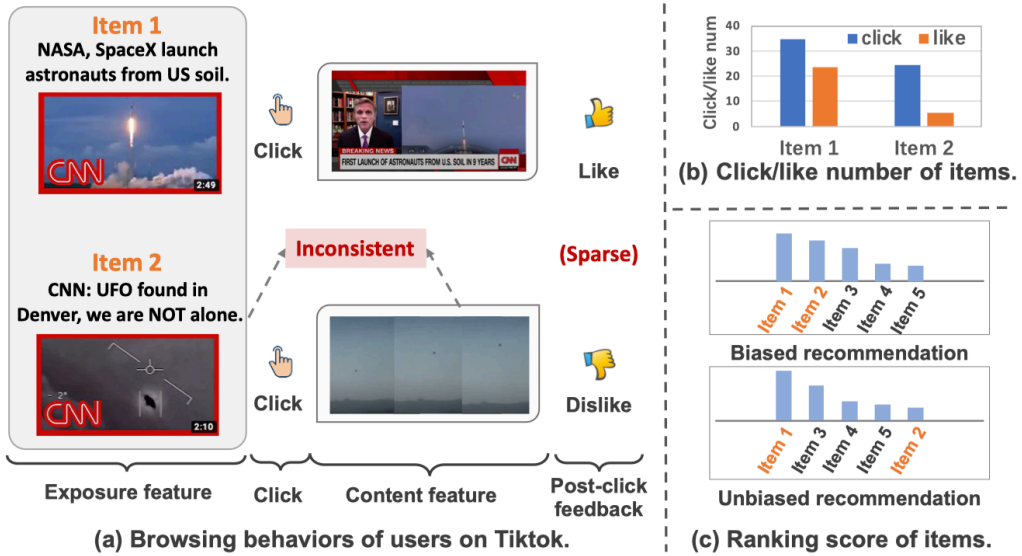
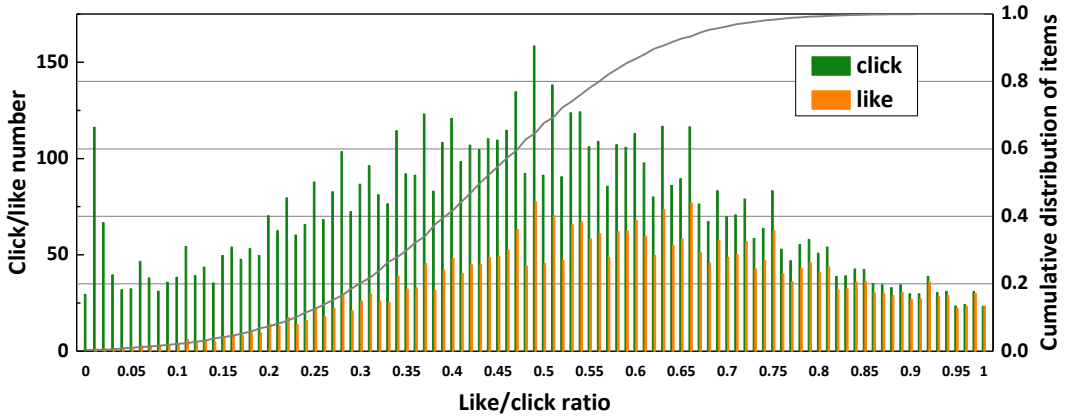


Fig. Statistics of clicks and likes on Tiktok dataset. Partly show the wide existence of clickbait issue.



Method

Counterfactual Recommendation (CR)

❖ Causal Graph

- A **causal graph** to describe the causal relationships between the features and user feedback.
- Exposure features and content features are fused into item features.
- **A direct shortcut** from exposure features to the prediction score: an item can be recommended purely because of its attractive title/cover.
- **Reference situation** denotes that the feature influence is null.

❖ NDE of exposure features on the prediction score

- Estimate the natural direct effect (NDE) in the counterfactual world, which **imagines** *what the prediction score would be if the item had only the exposure features.*

❖ CR inference:

- Reduce the direct effect of exposure features during inference.

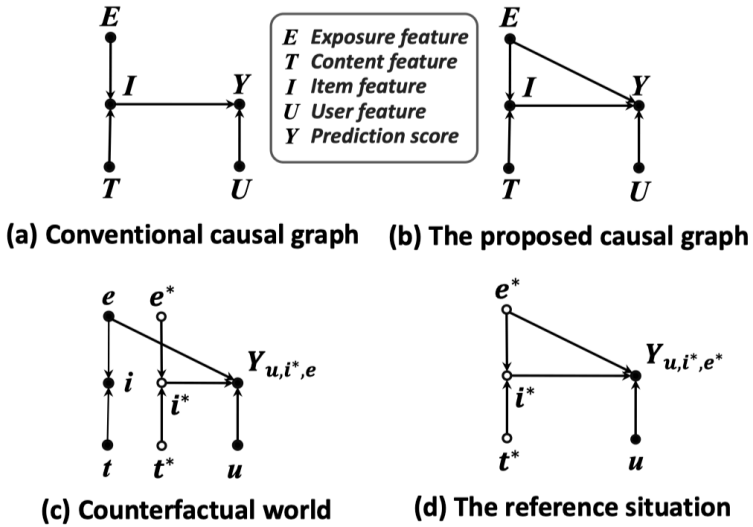
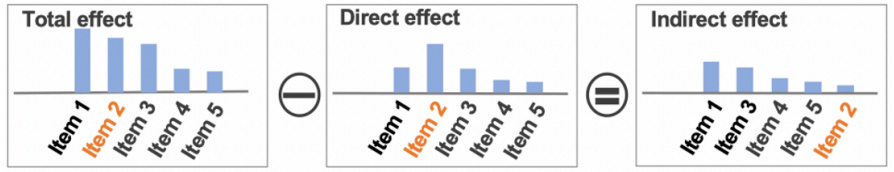
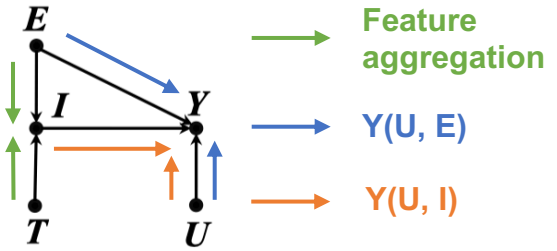


Figure 3: The causal graphs for conventional and counterfactual recommendations. ∗ denotes the reference values.



Method

Counterfactual Recommendation (CR)



- **CR framework:**
  1) Training: train a recommender to model the causal relations.
  2) Inference:
     a) Estimate NDE of E on Y.
     b) TE - NDE for inference.

- **Implementation**
  1. Backbone models to implement the causal graph.
  2. Two recommender models: one for $Y(U,I)$, another for $Y(U,E)$.
  3. A fusion function $f(\cdot)$ to learn the scoring function: $Y(U,I,E) = f(Y(U,I),Y(U,E))$.

$$\begin{aligned} Y_{u,i,e} &= Y(U=u, I=i, E=e) \\ &= f(Y_{u,i}, Y_{u,e}) \\ &= Y_{u,i} * \sigma(Y_{u,e}) \end{aligned}$$

$$\sum_{(u,i,\bar{Y}_{u,i})\in\bar{\mathcal{D}}} l(Y_{u,i,e}, \bar{Y}_{u,i}) + \alpha * l(Y_{u,e}, \bar{Y}_{u,i}), \quad (10)$$

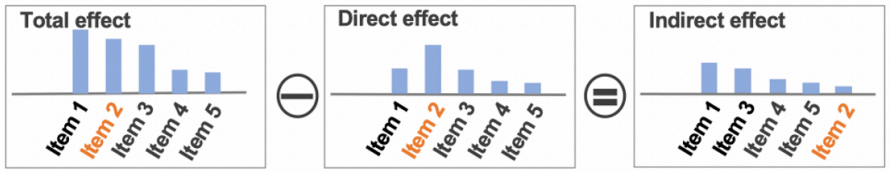
- **Training**
  - Optimize the recommenders with only clicks.
  - Multi-task training to learn the model parameters.

$$Y_{CR} = Y_{u,i,e} - Y_{u,i^*,e} = Y_{u,i,e} - f(c_u, Y_{u,e}) = Y_{u,i,e} - c_u * \sigma(Y_{u,e}).$$

c_u is the expectation constants for user u.

What the prediction score would be if the item had only the exposure features.

- **CR inference**
  1. NDE: estimate the prediction only based on exposure features.
  2. TE: original prediction based on all features.
  3. CR inference: TE – NDE to reduce the direct effect of E on Y.



Experiments

Experimental settings

- A model-agnostic framework. **Base model:** MMGCN *(Wei et al. 2019)*.
- **Datasets:** Adressa (news) & Tiktok (micro-video).
- **Evaluation**: evaluate the performance by post-click feedback (e.g., rating)**.**
- **Metric:** Precision@k, Recall@k, and NDCG@k.
- **Baselines:**
  - Training **without** post-click feedback.
    a) Normal Training (NT).
    b) Only using Content Feature for Training (CFT).
    c) Inverse Propensity Weighting (IPW). *(Liang et al. 2019)*
  - Training **with** post-click feedback.
    a) Cleaning Training (CT) which only uses the clicks end with likes as positive samples.
    b) Negative Reweighting (NR) which leverages post-click feedback to reweight negative samples. *(Wen et al. 2019)*
    c) Re-Rank (RR) the recommendation list of NT by the like/click ratio.

- *Liang et al. 2016. Causal inference for recommendation. In AUAI.*
- *Wen et al. 2019. Leveraging post-click feedback for content recommendations. In RecSys.*
- *Wei et al. 2019. MMGCN: Multi-modal Graph Convolution Network for Personalized Recommendation of Micro-video. In MM.*

Experiments

Overall Performance

Table 2: Top-K recommendation performance of compared methods on Tiktok and Adressa. %Improve. denotes the relative performance improvement of CR over NT. The best results are highlighted in bold. Stars and underlines denote the best results of the baselines with and without using additional post-click feedback during training, respectively.

Dataset	Tiktok						Adressa					
Metric	P@10	R@10	N@10	P@20	R@20	N@20	P@10	R@10	N@10	P@20	R@20	N@20
NT [50]	<u>0.0256</u>	<u>0.0357</u>	0.0333	<u>0.0231</u>	<u>0.0635</u>	0.0430	<u>0.0501</u>	<u>0.0975</u>	<u>0.0817</u>	<u>0.0415</u>	<u>0.1612</u>	<u>0.1059</u>
CFT [50]	0.0253	0.0356	<u>0.0339</u>	0.0226	0.0628	<u>0.0437</u>	0.0482	0.0942	0.0780	0.0405	0.1573	0.1021
IPW [27]	0.0230	0.0334	0.0314	0.0210	0.0582	0.0406	0.0419	0.0804	0.0663	0.0361	0.1378	0.0883
CT [50]	0.0217	0.0295	0.0294	0.0194	0.0520	0.0372	0.0493	0.0951	0.0799	0.0418*	0.1611	0.1051
NR [51]	0.0239	0.0346	0.0329	0.0216	0.0605	0.0424	0.0499	0.0970	0.0814	0.0415	0.1610	0.1058
RR	0.0264*	0.0383*	0.0367*	0.0231*	0.0635*	0.0430*	0.0521*	0.1007*	0.0831*	0.0415	0.1612*	0.1059*
CR	**0.0269**	**0.0393**	**0.0370**	**0.0242**	**0.0683**	**0.0476**	**0.0532**	**0.1045**	**0.0878**	**0.0439**	**0.1712**	**0.1133**
%Improve.	5.08%	10.08%	11.11%	4.76%	7.56%	10.70%	6.19%	7.18%	7.47%	5.78%	6.20%	6.99%

w/o post-click feedback

w/ post-click feedback

- **Observations:**
  - CFT and IPW perform worse than NT.
  - Post-click feedback could be helpful based on the performance of RR.
  - Proposed CR inference significantly recommends more satisfying items by mitigating clickbait issue.

Experiments

In-depth Analysis

❖ **Visualization of Recommendations w.r.t. Like/click Ratio.**

- We leverage post-click feedback (*e.g.,* ratings) to distinguish items: group by like/click ratio.
- **Items with low like/click ratio** are easy to have clickbait content.
- Proposed CR inference recommends less items with low like/click ratio, especially in [0, 0.4).

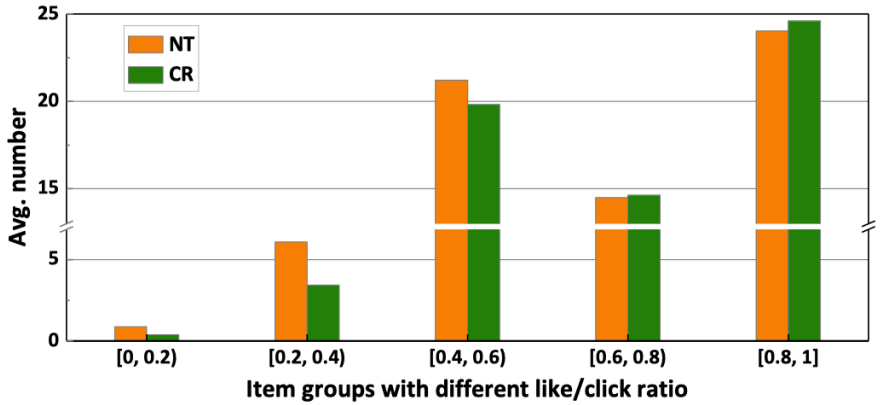


Fig. Visualization of the averaged recommendation frequencies of items. Note that items with low like/click ratios shouldn't be recommended.

Experiments

In-depth Analysis

❖ Effect of Dataset Cleanness.

- Study how the effectiveness of CR is influenced by the **"cleanness"** of the click data.

- **Settings:**
  1) **Rank** the items by the like/click ratio, and discard the items with high like/click ratios at a certain proportion;
  2) A larger discarding proportion leads to a dataset with more clicks with dislikes.

- **Observations:**
  - CR outperforms NT in all cases.
  - CR is significantly helpful in the scenarios with more noisy clicks.

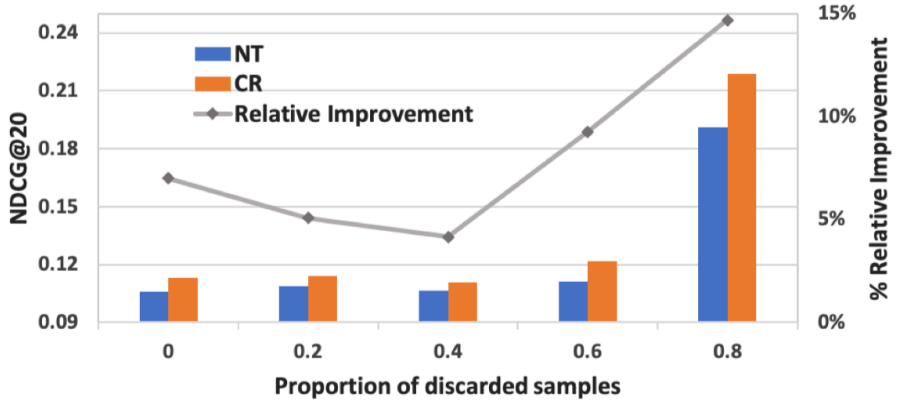


Fig. Performance comparison across the subsets of Adressa with different discarding proportions. A larger proportion indicates a higher percentage of the clicks that end with dislikes in the dataset.

Conclusion

❖ Summary

- Introduce an important but under-explored issue in recommendation: **clickbait issue**.
- **Inspect the causal relations** among the exposure features, content features, and predictions.
- Propose a framework of **counterfactual recommendation** for mitigating clickbait issue.

❖ Future work

- **A new task** of mitigating the clickbait issue that deserves our exploration.
- **Causality + Search/Recommendation:**
  - More comprehensive **causal graphs** with more fine-grained causal relations.
  - **Causal reasoning** over causal graph to mitigate other intrinsic biases and issues, such as causal intervention and counterfactual thinking.

Thank you !

Reference:

1. The book of why. Pearl et al. Basic Books, Inc., 2018.
2. Causality. J. Pearl, Cambridge university press, 2009.
3. Counterfactual VQA: A Cause-Effect Look at Language Bias. Niu et al. CVPR2021
4. Unbiased scene graph generation from biased training. Tang et al. CVPR2020
5. Learning to Recommend with Multiple Cascading Behaviors. Gao et al. TKDE2020
6. Chocolate consumption, cognitive function, and Nobel laureates. Messerli FH. N Engl J Med. 2012.
7. Causal inference for recommendation. Liang et al. AUAI2016.
8. Leveraging post-click feedback for content recommendations. Wen et al. RecSys2019.
9. MMGCN: Multi-modal Graph Convolution Network for Personalized Recommendation of Micro-video. Wei et al. MM2019.