Off-Policy Partial Feedback System Reward Estimation in Seznam.cz Web Search Engine

Pavel Procházka
Seznam.cz
pavel@prochazka.info

H2O Meetup
27.11.2019

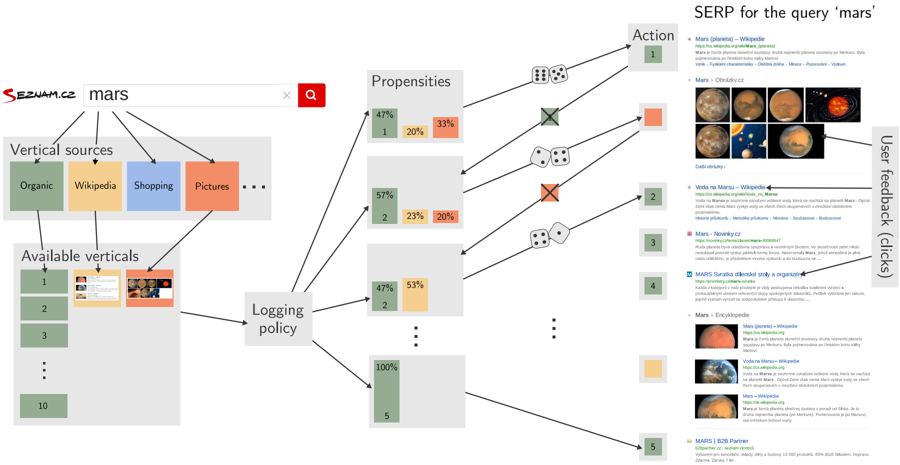
Outline

Outline

Seznam.cz

- Internet portal
- Over 20 services like
    - web search engine
    - web advertisement
    - email
    - maps
    - news, tv
    - and many more

Vertical Search Blending



Goals of the Talk

- partial feedback system abstraction
- performance evaluation in partial feedback systems
- introduction of basic off-policy evaluation methods
- example application in vertical search blending problem
- off-policy evaluation in your application?

Similar Possible Applications

- spell checker[a]
- recommender or advertising systems[b]
- counterfactual learning to rank[c]

[a] Lihong Li et al. "Counterfactual estimation and optimization of click metrics in search engines: A case study". In: *Proceedings of the 24th International Conference on WWW*. ACM. 2015, s. 929–934.

[b] Alexandre Gilotte et al. "Offline A/B Testing for Recommender Systems". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. WSDM '18. Marina Del Rey, CA, USA: ACM, 2018.

[c] Thorsten Joachims, Adith Swaminathan a Tobias Schnabel. "Unbiased learning-to-rank with biased feedback". In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM. 2017, s. 781–789.

Goals of the Talk

- partial feedback system abstraction
- performance evaluation in partial feedback systems
- introduction of basic off-policy evaluation methods
- example application in vertical search blending problem
- off-policy evaluation in your application?

Similar Possible Applications

- spell checker[a]
- recommender or advertising systems[b]
- counterfactual learning to rank[c]

[a] Lihong Li et al. “Counterfactual estimation and optimization of click metrics in search engines: A case study”. In: *Proceedings of the 24th International Conference on WWW*. ACM. 2015, s. 929–934.

[b] Alexandre Gilotte et al. “Offline A/B Testing for Recommender Systems”. In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. WSDM ’18. Marina Del Rey, CA, USA: ACM, 2018.

[c] Thorsten Joachims, Adith Swaminathan a Tobias Schnabel. “Unbiased learning-to-rank with biased feedback”. In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM. 2017, s. 781–789.

Outline

Contextual Bandits in Partial Feedback Systems

- **context** $x \sim P(x)$ (e.g. query, user id, ...)
- **action** $y \sim \pi(y|x)$ (e.g. chosen vertical)
- **reward function** $\delta(x, y)$ (e.g. click only for given (x, y))
- policy quality criterion $\rightarrow$ expected reward maximization

$$
\begin{aligned}
R(\pi) &= \mathrm{E}_{x\sim P(x), y\sim\pi(y|x)}[\delta(x, y)] \\
&= \sum_x \sum_y \delta(x, y)\pi(y|x)P(x)
\end{aligned}
$$

- partial feedback – no information about rewards for unseen actions
- assumption: stationary $\delta(x, y)$ and $P(x)$

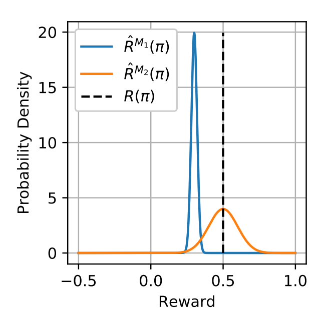
Expected Reward Estimate

- estimate bias:

$$\mathrm{E}\left[\hat{R}(\pi)\right] - R(\pi)$$

- estimate variance:

$$\mathrm{var}\left[\hat{R}(\pi)\right]$$



Online Evaluation – A/B Tests

- tested policy π_A deployment in production
- direct calculation

$$
\begin{aligned}
R(\pi_A) &= \mathrm{E}_{x \sim P(x), y \sim \pi_A(y|x)}[\delta(x, y)] \\
&\overset{(MC)}{\approx} \frac{1}{N} \sum_{i=1}^{N} \delta_i \\
&= \hat{R}^{AB}(\pi_A)
\end{aligned}
$$

- pros:
  - logged data and stationary assumption not needed
  - reliable and straightforward (no restriction from partial feedback)
- cons:
  - typically takes a long time
  - potential credit loss when testing a bad policy
  - production run dedicated purely to the test needed
  - expensive and difficult to scale

Online Evaluation Example

x	δ
"mars"	1
"H2O"	0
"cancer"	0
"shark"	1
"brexit"	0
"prague"	1

$$\hat{R}^{AB}(\pi_A) = \frac{1}{N}\sum_{i=1}^{N}\delta_i = \frac{1}{6}(1+0+0+1+0+1) = 0.5$$

Off-policy Evaluation

- desired reward estimation for alternative tested policy π_A
- based on offline logged data by policy π_0
- pros
    - cheap and scalable – no necessity for production deployment
    - possible to use the logged data for policy learning
- challenges
    - direct calculation of $\frac{1}{N}\sum_{i=1}^{N}\delta_i$ not possible due to partial feedback
    - validity issue – sanity checks
- methods solving the missing feedback
    - direct method
    - inverse propensity score (IPS) estimator
    - self normalized IPS estimator
    - doubly robust estimator

Direct Reward Estimation

- train a reward predictor $\hat{\delta}(x, y)$
- estimate the reward as

$$\begin{aligned} R(\pi_A) &= \mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\delta(x, y)] \\ &\overset{(DM)}{\approx} \mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\hat{\delta}(x, y)] \\ &\overset{(MC)}{\approx} \frac{1}{N}\sum_{i=1}^{N}\sum_{y}\pi_A(y|x_i)\hat{\delta}(x_i, y) \\ &= \hat{R}^{DM}(\pi_A) \end{aligned}$$

- world model $\hat{\delta}(x, y)$ required
- no logging policy feedback needed – data $\mathcal{D} = \{x_i\}_{i=1}^{N}$
- known to suffer from high bias

Direct Metod Evaluation Example

x	$\mathcal{A}(y)$	$\pi_A(y\vert x)$	$\hat{\delta}(x, y)$
”mars”	[pict, wiki, org]	[0, 1, 0]	$[0.2, \mathbf{0.1}, 0.3]$
”H2O”	[pict, wiki, org]	[0, 0, 1]	$[0.6, 0.7, \mathbf{0.6}]$
”cancer”	[pict, wiki, org]	[0, 0, 1]	$[0.2, 0.3, \mathbf{0.8}]$
”shark”	[pict, wiki, org]	[0, 1, 0]	$[0.1, \mathbf{0.9}, 0.1]$
”brexit”	[pict, wiki, org]	[1, 0, 0]	$[\mathbf{0.1}, 0.2, 0.3]$
”prague”	[pict, wiki, org]	[0, 1, 0]	$[0.99, \mathbf{0.9}, 0.99]$

$$
\begin{aligned}
\hat{R}^{DM}(\pi_A) &= \frac{1}{N}\sum_{i=1}^{N}\sum_{y}\pi_A(y|x_i)\hat{\delta}_i(x_i, y) \\
&= \frac{1}{6}(0.1 + 0.6 + 0.8 + 0.9 + 0.1 + 0.9) = 3.4/6
\end{aligned}
$$

IPS Reward Estimation

- data $\mathcal{D} = \{x_i, y_i, \delta_i, \pi_{0,i}\}_{i=1}^N$, $\delta_i = \delta(y_i, x_i)$ and $\pi_{0,i} = \pi_0(y_i|x_i)$

$$
\begin{aligned}
R(\pi_A) &= \mathrm{E}_{x\sim P(x), y\sim \pi_A(y|x)}[\delta(x,y)] \\
&\overset{(IPS)}{=} \mathrm{E}_{x\sim P(x), y\sim \pi_0(y|x)}\left[\frac{\pi_A(y|x)}{\pi_0(y|x)}\delta(x,y)\right] \\
&\overset{(MC)}{\approx} \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_{A,i}}{\pi_{0,i}}\delta_i \\
&= \hat{R}^{IPS}(\pi_A)
\end{aligned}
$$

- no need for $\hat{\delta}(y, x)$ – possible to estimate any logged feedback
- unbiased estimate typically with high variance
- estimate variance upper bound proportional to $1/\min(\pi_0)$
  - logging policy exploration required for IPS valid evaluation
  - more data for complex systems

IPS Evaluation Example

x	$\mathcal{A}(y)$	$\delta(x, y)$	$\pi_0(y\|x)$	$\pi_A(y\|x)$
"mars"	[**pict.**, wiki, org]	1	[0.2, 0.3, 0.5]	$[\mathbf{0}, 1, 0]$
"H2O"	[pict, **wiki**, org]	0	[0.1, 0.8, 0.1]	$[0, \mathbf{0}, 1]$
"cancer"	[pict, wiki, **org**]	0	$[0.1, 0.2, \mathbf{0.7}]$	$[0, 0, \mathbf{1}]$
"shark"	[pict, **wiki**, org]	1	$[0.4, \mathbf{0.4}, 0.2]$	$[0, \mathbf{1}, 0]$
"brexit"	[pict, wiki, **org**]	0	$[0.2, 0.2, 0.6]$	$[1, 0, \mathbf{0}]$
"prague"	[pict, **wiki**, org]	1	$[0.45, \mathbf{0.01}, 0.54]$	$[0, \mathbf{1}, 0]$

$$\hat{R}^{IPS}(\pi_A) = \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_{A,i}}{\pi_{0,i}}\delta_i$$

$$= \frac{1}{6}\left(\frac{1}{0.7}0 + \frac{1}{0.4}1 + \frac{1}{0.01}1\right) \approx 17.08$$

Off-Policy Evaluation Methods Properties

	Direct Method	**IPS**
approach	model the world	model the bias
bias	biased	unbiased
variance	typically low	high
data use from $\mathcal{D}$	only context $\{x_i\}_{i=1}^N$	use complete data
under/over fit	tends to under-fit on $\mathcal{D}$	tends to over-fit on $\mathcal{D}$
additional resources	$\hat{\delta}(x, y)$	—
improvements	DRO	SNIPS, DRO

Self Normalized Inverse Propensity Score Estimator

- motivation – IPS with reduced variance
- idea[a] – norm (regularize) IPS with a constant C

$$\hat{R}^{\text{SNIPS}} = \frac{\hat{R}^{IPS}}{C}, \quad C = \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_{A,i}}{\pi_{0,i}}$$

- add small bias, but reduce variance
- possible to estimate any feedback metric (reward)
- no need for $\hat{\delta}(x, y)$
- sanity check[b]: denominator $C = \frac{1}{N}\sum_{i=1}^{N} c_i \rightarrow 1$

[a] Adith Swaminathan a Thorsten Joachims. “The self-normalized estimator for counterfactual learning”. In: *Advances in Neural Information Processing Systems*. 2015, s. 3231–3239.

[b] Damien Lefortier et al. “Large-scale Validation of Counterfactual Learning Methods: A Test-Bed”. In: *CoRR* abs/1612.00367 (2016). arXiv: `1612.00367`.

SNIPS Evaluation Example

x	$\mathcal{A}(y)$	$\delta(x, y)$	$\pi_0(y\|x)$	$\pi_A(y\|x)$
"mars"	[**pict.**, wiki, org]	1	[0.2, 0.3, 0.5]	[0, 1, 0]
"H2O"	[pict, **wiki**, org]	0	[0.1, 0.8, 0.1]	[0, 0, 1]
"cancer"	[pict, wiki, **org**]	0	[0.1, 0.2, **0.7**]	[0, 0, 1]
"shark"	[pict, **wiki**, org]	1	[0.4, **0.4**, 0.2]	[0, 1, 0]
"brexit"	[pict, wiki, **org**]	0	[0.2, 0.2, 0.6]	[1, 0, 0]
"prague"	[pict, **wiki**, org]	1	[0.45, **0.01**, 0.54]	[0, 1, 0]

$$C = \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_{A,i}}{\pi_{0,i}}$$

$$= \frac{1}{6}\left(\frac{1}{0.7} + \frac{1}{0.4} + \frac{1}{0.01}\right) \approx 17.32 \gg 1$$

$$\hat{R}^{\mathrm{SNIPS}}(\pi_A) = \frac{\hat{R}^{IPS}(\pi_A)}{C} = \frac{17.08}{17.32} \approx 0.99$$

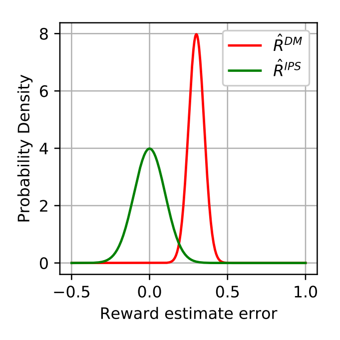
Doubly Robust Estimator

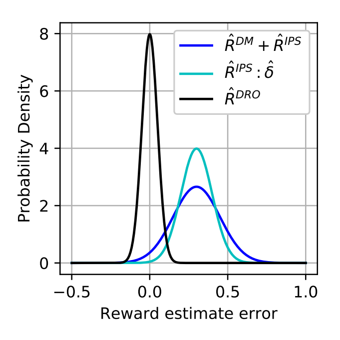
- motivation – fully utilize both $\mathcal{D}$ and $\hat{\delta}(x, y)$

$$\begin{aligned}
R(\pi_A) &= \mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\delta(x,y)] \\
&= \mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\delta(x,y) - \hat{\delta}(x,y) + \hat{\delta}(x,y)] \\
&= \underbrace{\mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\hat{\delta}(x,y)]}_{\text{DirectMethod}} \\
&+ \underbrace{\mathrm{E}_{x\sim P(x), y\sim\pi_A(y|x)}[\delta(x,y) - \hat{\delta}(x,y)]}_{\text{IPS}} \\
&\overset{(MC)}{\approx} \frac{1}{N}\sum_{i=1}^{N}\sum_{y}\pi_A(y|x_i)\hat{\delta}(x_i, y) \\
&+ \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_A(y_i|x_i)}{\pi_0(y_i|x_i)}(\delta(x_i, y_i) - \hat{\delta}(x_i, y_i)) \\
&= \hat{R}^{DRO}(\pi_A)
\end{aligned}$$

Doubly Robust Estimator

- use all available information[a] – both $\mathcal{D}$ and $\hat{\delta}(x, y)$
- intuition





[a] Miroslav Dudik, John Langford a Lihong Li. “Doubly Robust Policy Evaluation and Learning”. In: *Proceedings of the 28th ICML*. ICML'11. Bellevue, Washington, USA: Omnipress, 2011, s. 1097–1104. ISBN: 978-1-4503-0619-5.

Doubly Robust Evaluation Example

x	$\mathcal{A}(y)$	$\delta(x, y)$	$\pi_0(y\|x)$	$\pi_A(y\|x)$	$\hat{\delta}(x, y)$
"mars"	[**pict.**, wiki, org]	1	[0.2, 0.3, 0.5]	[0, 1, 0]	[0.2, 0.1,0.3]
"H2O"	[pict, **wiki**, org]	0	[0.1, 0.8, 0.1]	[0, 0, 1]	[0.6, 0.7,0.6]
"cancer"	[pict, wiki, **org**]	0	[0.1, 0.2, **0.7**]	[0, 0, 1]	[0.2, 0.3, **0.8**]
"shark"	[pict, **wiki**, org]	1	[0.4, **0.4**, 0.2]	[0, 1, 0]	[0.1, **0.9**, 0.1]
"brexit"	[pict, wiki, **org**]	0	[0.2, 0.2, 0.6]	[1, 0, 0]	[0.1, 0.2, 0.3]
"prague"	[pict, **wiki**, org]	1	[0.45, **0.01**, 0.54]	[0, 1, 0]	[0.99, **0.9**, 0.99]

$$
\begin{aligned}
\hat{R}^{DRO}(\pi_A) &= \hat{R}^{IPS}(\pi_A) + \hat{R}^{DM}(\pi_A) - \frac{1}{N}\sum_{i=1}^{N}\frac{\pi_{A,i}}{\pi_{0,i}}\hat{\delta}_i \\
&= \hat{R}^{IPS}(\pi_A) + \hat{R}^{DM}(\pi_A) - \frac{1}{6}\left(\frac{1}{0.7}0.8 + \frac{1}{0.4}0.9 + \frac{1}{0.01}0.9\right) \\
&\approx 2.32
\end{aligned}
$$

Off-Policy Learning

- reward estimation as optimization objective
- direct method
  1. train reward predictor for all possible actions
  2. choose the action with the highest predicted reward
- IPS (counterfactual approach)[a]
  - the best action selection → classification task
  - much easier task than regression
- doubly robust
- methods (IPS, DM, DRO) implemented in Vowpal Wabbit

```
vw -d train.dat --cb 4 --cb_type ips
```

https://github.com/VowpalWabbit/vowpal_wabbit/wiki/Logged-Contextual-Bandit-Example

[a] Thorsten Joachims a Adith Swaminathan. “SIGIR Tutorial on Counterfactual Evaluation and Learning for Search, Recommendation and Ad Placement”. In: *Proceedings of the 39th International ACM SIGIR*. ACM, 2016.

Outline

Per-Position Setup

- **context** $x \sim P(x)$ – SERP features + results on positions above
- **reward function** $\delta(x, y)$ – click
- **action** $y \sim \pi(y|x)$ – chosen vertical at given position
- **propensity** – directly logged by policy
- low number of actions – expected reasonable propensity values
- not clear interpretation[a]

[a] Pavel Prochazka et al. “Vertical Search Blending: A Real-world Counterfactual Dataset”. In: *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR’19. Paris, France: ACM, 2019, s. 1237–1240. URL: http://doi.acm.org/10.1145/3331184.3331345.

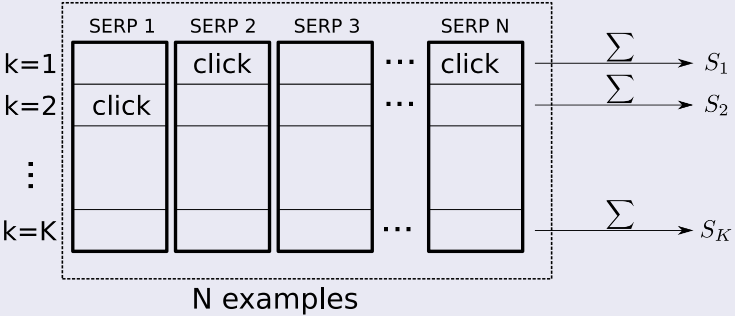
Per-SERP Setup

- **context** $x \sim P(x)$ – SERP features
- **reward function** $\delta(x, y)$ – click, ndcg, ...
- **action** $y \sim \pi(y|x)$ – complete SERP composition
- **propensity** chain rule: $\pi_0(y|x) = \prod_{k=1}^{K} q_k$, q_k logged propensity for the k-th position
- SERP-length K: complexity-expressiveness trade-off – unreliable estimate of target metric versus reliable estimate of partial metric

Counterfactual Framework Application

	Fixed position	**Complete SERP**
action	result at given position a_k	SERP $\boldsymbol{b}_K = (a_1, \dots, a_K)$
propensity q, π	$q_k := p(a_k \mid a_1, \dots, a_{k-1}, \chi) = p(a_k \mid \boldsymbol{b}_{k-1}, \chi)$	$\pi_K := p(\boldsymbol{b}_K \mid \chi) = \prod_{k=1}^{K} q_k$
feedback δ	click at given position	click to SERP, NDCG, etc.
context	SERP features χ and results on the first $k-1$ positions $\boldsymbol{b}_{k-1}$	SERP features χ
number of actions	low – typically a few possible actions	increasing significantly with SERP length K
reward interpretation	CTR at the first position, almost meaningless otherwise	expresses target SERP business metrics
purpose	training models	evaluation of target metric

Vertical Search Blending Induced Sanity Check



$$R_k^{\mathrm{CTR}} = \frac{S_1+S_2+\cdots+S_k}{N} \rightarrow \text{sanity check: } \hat{R}_1^{\mathrm{CTR}} \leq \hat{R}_2^{\mathrm{CTR}} \leq \cdots \leq \hat{R}_K^{\mathrm{CTR}}$$

Evaluation Setup

- **training** – DM, DRO, IPS using Vowpal Wabbit in per position setup
- per-position SERP composition
- **evaluation** – SNIPS estimate of trained models + logging and random policies
- **sanity checks** – non-decreasing CTR, $C \to 1$
- **evaluated metrics** – CTR, average NDCG
- trade-off parameter K

Result – SNIPS Estimates of Metrics on SERP

Policy	K=1			K=2			K=3		
	C	$\hat{R}_1^{\mathrm{CTR}}$	$\hat{R}_1^{\mathrm{NDCG}}$	C	$\hat{R}_2^{\mathrm{CTR}}$	$\hat{R}_2^{\mathrm{NDCG}}$	C	$\hat{R}_3^{\mathrm{CTR}}$	$\hat{R}_3^{\mathrm{NDCG}}$
DM	0.922	0.435	0.382	1.046	0.523	0.435	3.646	$0.487 < \hat{R}_2^{\mathrm{CTR}}$	0.360
DR	0.837	0.445	0.393	0.902	0.535	0.449	2.971	$0.487 < \hat{R}_2^{\mathrm{CTR}}$	0.364
IPS	0.020	0.408	0.343	0.013	0.639	0.509	0.015	$0.632 < \hat{R}_2^{\mathrm{CTR}}$	0.493
Random	0.480	0.404	0.351	0.406	0.474	0.377	0.626	0.499	0.368
Logging	1.000	0.433	0.379	1.000	0.526	0.434	1.000	0.571	0.458
YM	?	?	?	?	?	?	?	?	?

Concluding Remarks

- off policy evaluation as viable alternative / complement to A/B testing
- policy evaluation is not for free
  - A/B testing
  - needed randomization (exploration) of logging policy
- validity issue and importance of sanity checks
- outlined off-policy learning
- vertical search blending data-set available at

  `https://github.com/seznam/vertical-search-blending-dataset`

References

Miroslav Dudik, John Langford a Lihong Li. “Doubly Robust Policy Evaluation and Learning”. In: *Proceedings of the 28th ICML*. ICML’11. Bellevue, Washington, USA: Omnipress, 2011, s. 1097–1104. ISBN: 978-1-4503-0619-5.

Alexandre Gilotte et al. “Offline A/B Testing for Recommender Systems”. In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. WSDM ’18. Marina Del Rey, CA, USA: ACM, 2018.

Thorsten Joachims a Adith Swaminathan. “SIGIR Tutorial on Counterfactual Evaluation and Learning for Search, Recommendation and Ad Placement”. In: *Proceedings of the 39th International ACM SIGIR*. ACM, 2016.

Thorsten Joachims, Adith Swaminathan a Tobias Schnabel. “Unbiased learning-to-rank with biased feedback”. In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM. 2017, s. 781–789.

Damien Lefortier et al. “Large-scale Validation of Counterfactual Learning Methods: A Test-Bed”. In: *CoRR* abs/1612.00367 (2016). arXiv: `1612.00367`.

Lihong Li et al. “Counterfactual estimation and optimization of click metrics in search engines: A case study”. In: *Proceedings of the 24th International Conference on WWW*. ACM. 2015, s. 929–934.

Pavel Prochazka et al. “Vertical Search Blending: A Real-world Counterfactual Dataset”. In: *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR’19. Paris, France: ACM, 2019, s. 1237–1240. URL: `http://doi.acm.org/10.1145/3331184.3331345`.

Adith Swaminathan a Thorsten Joachims. “The self-normalized estimator for counterfactual learning”. In: *Advances in Neural Information Processing Systems*. 2015, s. 3231–3239.

https://kariera.seznam.cz/

Thanks for attention

Comments & Questions?