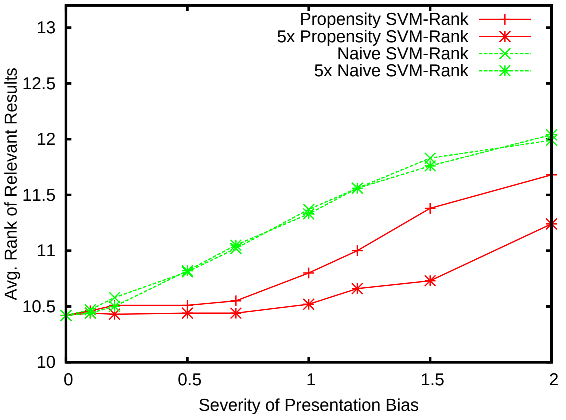

Figure 2: Test set performance for Propensity SVM-Rank and Naive SVM-Rank as presentation bias becomes more severe in terms of η ($n = 45K$ and $n = 225K$, $\epsilon_- = 0$).

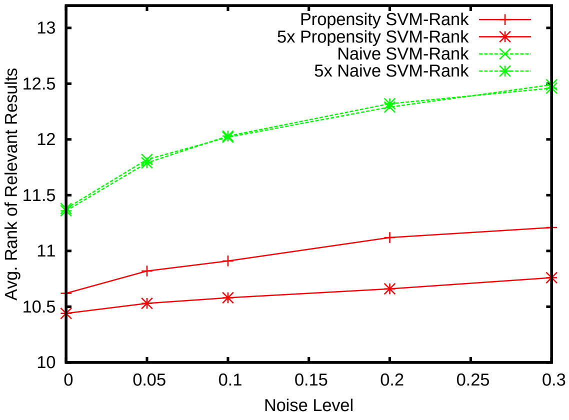

Figure 3: Test set performance for Propensity SVM-Rank and Naive SVM-Rank as the noise level increases in terms of ϵ_- ($n = 170K$ and $n = 850K$, $\eta = 1$).

and noise $\epsilon_- = 0.1$. For small datasets, results are averaged over 5 draws of the click data.

With increasing amounts of click data, Propensity SVM-Rank approaches the skyline performance of the full-information SVM-Rank trained on the complete training set of manual ratings without noise. This is in stark contrast to Naive SVM-Rank which fails to account for the bias in the data and does not reach this level of performance. Furthermore, Naive SVM-Rank cannot make effective use of additional data and its learning curve is essentially flat. This is consistent with the theoretical insight that estimation error in Naive SVM-Rank's empirical risk $\hat{R}(S)$ is dominated by asymptotic bias due to biased clicks, which does not decrease with more data and leads to suboptimal learning. The unbiased risk estimate $\hat{R}_{IPS}(S)$ of Propensity SVM-Rank, however, has estimation error only due to finite sample variance, which is decreased by more data and leads to consistent learning.

While unbiasedness is an important property when click data is plenty, the increased variance of $\hat{R}_{IPS}(S)$ can be a drawback for small datasets. This can be seen in Figure 1, where Naive SVM-Rank outperforms Propensity SVM-Rank for small datasets. This can be remedied using techniques like "propensity clipping" [23], where small propensities are clipped to some threshold value τ to trade bias for variance.

$$\hat{R}_{CIPS}(S) = \frac{1}{n}\sum_{\boldsymbol{x}_i}\sum_{y\in S(\boldsymbol{x}_i)}\frac{\mathrm{rank}(y|S(\boldsymbol{x}_i))\cdot \mathrm{r}_i(y)}{\max\{\tau, Q(o_i(y)\!=\!1|\boldsymbol{x}_i,\bar{\boldsymbol{y}}_i,\mathrm{r}_i)\}}.$$

Figure 1 shows the learning curve of Propensity SVM-Rank with clipping, cross-validating both the clipping threshold τ and C. Clipping indeed improves performance for small datasets. While $\tau = 1$ is equivalent to Naive SVM-Rank, the validation set is too small (and hence, the finite sample error of the validation performance estimate too high) to reliably select this model in every run. In practice, however, we expect click data to be plentiful such that lack of training data is unlikely to be a persistent issue.

7.3 How much presentation bias can be tolerated?

We now vary the severity of the presentation bias via η to understand its impact on Propensity SVM-Rank. Figure 2 shows that inverse propensity weighting is beneficial whenever substantial bias exists. Furthermore, increasing the amount of training data by a factor of 5 leads to further improvement for the Propensity SVM-Rank, while the added training data has no effect on Naive SVM-Rank. This is consistent with our arguments from Section 4 – more training data does not help when bias dominates estimation error, but it can reduce estimation error from variance in the unbiased risk estimate of Propensity SVM-Rank.

7.4 How robust are the methods to click noise?

Figure 3 shows that Propensity SVM-Rank also enjoys a substantial advantage when it comes to noise. When increasing the noise level in terms of ϵ_- from 0 up to 0.3 (resulting in click data where 59.8% of all clicks are on irrelevant documents), Propensity SVM-Rank increasingly outperforms Naive SVM-Rank. And, again, the unbiasedness of the empirical risk estimate allows Propensity SVM-Rank to benefit from more data.

7.5 How robust is Propensity SVM-Rank to misspecified propensities?

So far all experiments have assumed that Propensity SVM-Rank has access to accurate propensities. In practice, however, propensities need to be estimated and are subject to model assumptions. We now evaluate how robust Propensity SVM-Rank is to misspecified propensities. Figure 4 shows the performance of Propensity SVM-Rank when the training data is generated with $\eta = 1$, but the propensities used by Propensity SVM-Rank are misspecified using the η given in the x-axis of the plot. The plot shows that even misspecified propensities can give substantial improvement over naively ignoring the bias, as long as the misspecification is "conservative" – i.e., overestimating small propensities is tolerable (which happens when $\eta < 1$), but underestimating small propensities can be harmful (which happens when $\eta > 1$). This is consistent with theory, and clipping is one particular way of overestimating small propensities that can even improve performance. Overall, we conclude that even a mediocre propensity model can improve over the naive approach – after all, the naive approach can be thought of as a particularly poor propensity model that implicitly assumes no presentation bias and uniform propensities.

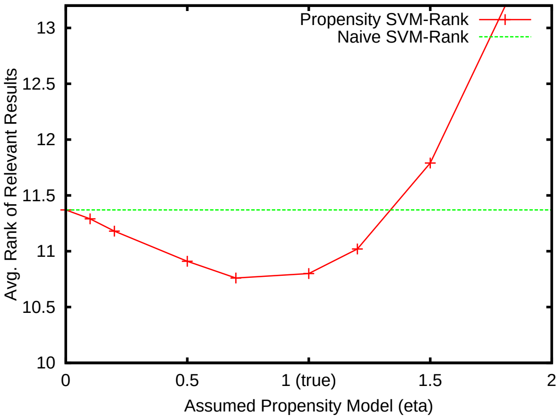

Figure 4: Test set performance for Propensity SVM-Rank and Naive SVM-Rank as propensities are misspecified (true $\eta = 1$, $n = 170K$, $\epsilon_- = 0.1$).

7.6 Real-World Experiment

We now examine the performance of Propensity SVM-rank when trained on real-world click logs and deployed in a live search engine for scientific articles [anonymized for submission]. The search engine uses a linear scoring function as outlined in Section 6. Query-document features $\phi(\boldsymbol{x}, y)$ are represented by a $1000-$dimensional vector, and the production ranker used for collecting training clicks employs a hand-crafted weight vector w (denoted Prod). Observed clicks on rankings served by this ranker over a period of 21 days provide implicit feedback data for LTR as outlined in Section 6.

To estimate the propensity model, we consider the simple position-based model of Section 5.1 and we collect new click data via randomized interventions for 7 days as outlined in Section 5.3 with landmark rank $k = 1$. Before presenting the ranking, we take the top-ranked document and swap it with the document at a uniformly at random chosen rank $j \in \{1, \ldots 21\}$. The ratio of observed click-through rates (CTR) on the formerly top-ranked document now at position j vs. its CTR at position 1 gives a noisy estimate of p_j/p_1 in the position-based click model. We additionally smooth these estimates by interpolating with the overall observed CTR at position j (normalized so that $CTR@1 = 1$). This yields p_r that approximately decay with rank r with the smallest $p_r \simeq 0.12$. For $r > 21$, we impute $p_r = p_{21}$.

We partition the click-logs into a train-validation split: the first 16 days are the train set and provide 5437 click-events for SVM-rank, while the remaining 5 days are the validation set with 1755 click events. The hyper-parameter C is picked via cross validation. Analogous to Section 7.1, we use the IPS estimator for Propensity SVM-Rank, and naive estimator with $Q(o(y) = 1|\boldsymbol{x}, \bar{\boldsymbol{y}}, \mathrm{r}) = 1$ for Naive SVM-Rank. With the best hyper-parameter settings, we re-train on all 21 days worth of data to derive the final weight vectors for either method.

We fielded these learnt weight vectors in two online interleaving experiments [2], the first comparing Propensity SVM-Rank against Prod and the second comparing Propensity SVM-Rank against Naive SVM-Rank. The results are summarized in Table 1. We find that Propensity SVM-Rank significantly outperforms the hand-crafted production ranker that was used to collect the click data for training (two-tailed binomial sign test $p = 0.001$ with relative risk 0.71 compared to null hypothesis). Furthermore, Propensity SVM-Rank similarly outperforms Naive SVM-Rank, demonstrating that even a simple propensity model provides benefits on real-world data (two-tailed binomial sign test $p = 0.006$ with relative risk 0.77 compared to null hypothesis). Note that Propensity SVM-Rank not only significantly, but also substantially outperforms both other rankers in terms of effect size – and the synthetic data experiments suggest that additional training data will further increase its advantage.

Table 1: Per-query balanced interleaving results for detecting relative performance between the hand-crafted production ranker used for click data collection (Prod), Naive SVM-Rank and Propensity SVM-Rank.

	Propensity SVM-Rank		
Interleaving Experiment	wins	loses	ties
against Prod	87	48	83
against Naive SVM-Rank	95	60	102

8. CONCLUSIONS

This paper introduced a principled approach for learning-to-rank under biased feedback data. Drawing on counterfactual modeling techniques from causal inference, we present a theoretically sound Empirical Risk Minimization framework for LTR. We instantiate this framework with a Propensity-Weighted Ranking SVM, and provide extensive empirical evidence that the resulting learning method is robust to selection biases, noise, and model misspecification. Furthermore, our real-world experiments on a live search engine show that the approach leads to substantial retrieval improvements, without any heuristic or manual interventions in the learning process.

9. FUTURE RESEARCH

Beyond the specific learning methods and propensity models we propose, this paper may have even bigger impact for its theoretical contribution of developing the general counterfactual model for LTR, thus articulating the key components necessary for LTR under biased feedback. First, the insight that propensity estimates are crucial for ERM learning opens a wide area of research on designing better propensity models. Second, the theory demonstrates that LTR methods should optimize propensity-weighted ERM objectives, raising the question of which other learning methods beyond the Ranking SVM can be adapted to the Propensity ERM approach. Third, we conjecture that a Propensity ERM approach can be developed also for pointwise LTR methods using techniques from [19], and possibly even for listwise LTR.

Beyond learning from implicit feedback, propensity-weighted ERM techniques may prove useful even for optimizing offline IR metrics on manually annotated test collections. First, they can eliminate pooling bias, since the use of sampling during judgment elicitation puts us in a controlled setting where propensities are known (and can be optimized [19]) by design. Second, propensities estimated via click models can enable click-based IR metrics like click-DCG to better correlate with test set DCG.

10. ACKNOWLEDGMENTS

This work was supported in part through NSF Awards IIS-1247637, IIS-1513692, IIS-1615706, and a gift from Bloomberg. We thank Maarten de Rijke, Alexey Borisov, Artem Grotov, and Yuning Mao for valuable feedback and discussions.

11. REFERENCES

[1] A. Borisov, I. Markov, M. de Rijke, and P. Serdyukov. A neural click model for web search. In *Proceedings of the 25th International Conference on World Wide Web*, pages 531–541, 2016.

[2] O. Chapelle, T. Joachims, F. Radlinski, and Y. Yue. Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems (TOIS)*, 30(1):6:1–6:41, 2012.

[3] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *International Conference on World Wide Web (WWW)*, pages 1–10. ACM, 2009.

[4] A. Chuklin, I. Markov, and M. de Rijke. *Click Models for Web Search*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2015.

[5] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *International Conference on Web Search and Data Mining (WSDM)*, pages 87–94. ACM, 2008.

[6] K. Hofmann, A. Schuth, S. Whiteson, and M. de Rijke. Reusing historical interaction data for faster online learning to rank for ir. In *International Conference on Web Search and Data Mining (WSDM)*, pages 183–192, 2013.

[7] D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952.

[8] G. Imbens and D. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015.

[9] T. Joachims. Optimizing search engines using clickthrough data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 133–142, 2002.

[10] T. Joachims. Training linear SVMs in linear time. In *ACM SIGKDD International Conference On Knowledge Discovery and Data Mining (KDD)*, pages 217–226, 2006.

[11] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Transactions on Information Systems (TOIS)*, 25(2), April 2007.

[12] L. Li, W. Chu, J. Langford, and X. Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *International Conference on Web Search and Data Mining (WSDM)*, pages 297–306, 2011.

[13] R. J. A. Little and D. B. Rubin. *Statistical Analysis with Missing Data*. John Wiley, 2002.

[14] T.-Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, Mar. 2009.

[15] K. Raman and T. Joachims. Learning socially optimal information systems from egoistic users. In *European Conference on Machine Learning (ECML)*, pages 128–144, 2013.

[16] K. Raman, T. Joachims, P. Shivaswamy, and T. Schnabel. Stable coactive learning via perturbation. In *International Conference on Machine Learning (ICML)*, pages 837–845, 2013.

[17] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: Estimating the click-through rate for new ads. In *International Conference on World Wide Web (WWW)*, pages 521–530. ACM, 2007.

[18] P. R. Rosenbaum and D. B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

[19] T. Schnabel, A. Swaminathan, P. Frazier, and T. Joachims. Unbiased comparative evaluation of ranking functions. In *ACM International Conference on the Theory of Information Retrieval (ICTIR)*, 2016.

[20] T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims. Recommendations as treatments: Debiasing learning and evaluation. In *International Conference on Machine Learning (ICML)*, 2016.

[21] A. Schuth, H. Oosterhuis, S. Whiteson, and M. de Rijke. Multileave gradient descent for fast online learning to rank. In *International Conference on Web Search and Data Mining (WSDM)*, pages 457–466, 2016.

[22] K. Sparck-Jones and C. J. V. Rijsbergen. Report on the need for and provision of an “ideal” information retrieval test collection. Technical report, University of Cambridge, 1975.

[23] A. L. Strehl, J. Langford, L. Li, and S. Kakade. Learning from logged implicit exploration data. In *Proceedings of the 24th Annual Conference on Neural Information Processing Systems*, pages 2217–2225, 2010.

[24] A. Swaminathan and T. Joachims. Batch learning from logged bandit feedback through counterfactual risk minimization. *Journal of Machine Learning Research (JMLR)*, 16:1731–1755, Sep 2015. Special Issue in Memory of Alexey Chervonenkis.

[25] V. Vapnik. *Statistical Learning Theory*. Wiley, Chichester, GB, 1998.

[26] L. Wang, J. J. Lin, and D. Metzler. A cascade ranking model for efficient ranked retrieval. In *ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 105–114. ACM, 2011.

[27] X. Wang, M. Bendersky, D. Metzler, and M. Najork. Learning to rank with selection bias in personal search. In *ACM Conference on Research and Development in Information Retrieval (SIGIR)*. ACM, 2016.

[28] Y. Wang, D. Yin, L. Jie, P. Wang, M. Yamada, Y. Chang, and Q. Mei. Beyond ranking: Optimizing whole-page presentation. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM ’16, pages 103–112, 2016.

[29] Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *International Conference on Machine Learning (ICML)*, pages 151–159, 2009.