Scientific Report for ELIAS Exchange Visit Grant

Online Evaluation for Online Learning to Rank

Anne Schuth, anne.schuth@uva.nl
ISLA, University of Amsterdam, The Netherlands
Reference Number: 4683

May 26, 2014

I hereby report on my recent 3 month visit to Yandex in Moscow that was funded by the ELIAS Network Programme. At Yandex I collaborated with the research group lead by dr. Pavel Serdyukov.

Purpose of Visit

My PhD thesis' working title is "Learning from Simulations, Users and Annotators: Online Learning to Rank for Information Retrieval." It spans several research areas including reinforcement learning, user modeling, learning to rank, and online evaluation. During previous years, I have worked on all these areas but so far, I have only applied them to a setting with *simulations* of users (Hofmann et al., 2013; Chuklin et al., 2013; Hofmann et al., 2014; Schuth et al., 2014b, 2013, 2014a). At Yandex, I was able to take my research to the next level by deploying my ideas in a setting with real users. Yandex, being among the major search engines in the world, turned out to be the ideal place to do so.

State-of-the-art *information retrieval* (IR) systems such as Yandex have many parameters, such as weights for their ranking features, that need to be optimized. Finding the optimal parameter settings is a crucial task for a search engine as it directly reflects the engine's quality. This task, however, is not only crucial, it is also a hard task because the parameter space is enormous as it typically ranges into hundreds of dimensions. Moreover, the requirements and expectations of both users and system designers are constantly changing, which demands an adaptive approach to parameter optimization. IR systems are in the position where they can optimize their models based on interactions with users to adapt to the needs of these same users. The large amount of e.g., click data that can be collected in search settings, paves the way for online learning to rank for IR systems. *Online learning to rank* (OL2R) is the process of learning these optimal parameters from this source of implicit user feedback such as clicks.

Typically (Yue and Joachims, 2009), OL2R starts with what is currently known to be the best parameter setting, which we call the *exploitative ranker*.

From there, the algorithm takes a small exploratory step in a random (Yue and Joachims, 2009) or guided (Hofmann et al., 2013) direction in parameter space to try whether this direction constitutes an improvement. The exploitative ranker is compared to this perturbation, which we call the *exploratory ranker*. This comparison is performed using an *online evaluation* method. It has been shown that *interleaved comparison* methods, such as TeamDraft are best suited for online evaluation (Radlinski et al., 2008). This method works as follows. When a user issues a query, documents are ranked using both the exploitative and exploratory ranker. The resulting two rankings are interleaved and shown to the user. The clicks from the user, on documents from the interleaved ranking, are then interpreted by the interleaving method to decide on the winner. If the exploratory ranker wins the comparisons, the weights of the exploitative ranker will be updated slightly towards those of the exploratory ranker. In principle, this process repeats indefinitely resulting in an adaptive optimization method.

So far, OL2R has been applied in large scale commercial settings, with feedback from user, to ad-placement. However, to the best of my knowledge, it has never been applied to web search (other then in a simulated environment). One of the main reasons to *not* apply OL2R to such a crucial product as web search, is that the updates of weights are based on *local* evidence that it constitutes an improvement; evidence from a single query. It is thus not guaranteed that updates also constitute a *global* improvement; an improvement for all users or all queries.

Related to these concerns, I was interested in investigating the following research questions while at Yandex.

1. *How we can control (in order to trust) an OL2R algorithm enough to allow it to run in such a critical setting?*
   My ideas towards answering this question include the following. I would investigate how similar (or dissimilar) the ranking systems are that are currently being allowed in the interleaving experiments at Yandex. In other words, to what extent variation in user experience is tolerated? By measuring, for instance, the rank-correlation of rankings produced for the same query by different systems, this can be quantified. Then, one could design the system such that it only allows exploratory rankers within a bounded subspace of the whole parameter space that guarantees a minimal rank correlation for a sample of queries. Additionally, confidence bounds can be placed on the decisions of the interleaved comparisons. After comparing the exploitative ranker and exploratory ranker on multiple queries, it would then be possible to say to what degree we can be confident in the correctness of the decision on the winner.

2. *How can we guarantee a maximum degree and extent of degeneration of user experience?*
   While, using the techniques described above, an OL2R system is guaranteed to not wander of in unwanted directions too far, there is still no guaranty of global improvement. To answer this second question, a separate quality control mechanism should be in place that, for instance, monitors absolute click metrics. Or, one that constantly evaluates the exploitative ranker on an annotated dataset.

3. *How much can we still learn, given the restrictions imposed on the explo-*

ration?
Assuming that answers to the previous two questions will place restrictions on the exploration, I am interested in what we can expect to learn and how long it takes to learn this. There will be a trade-off between potential but immediate user experience degeneration and rewards in the future in the form of better user experience. How big should the exploration step for each query be, and how many queries do we then need, given the quality of the feedback, to arrive at a significantly better ranker? Can we give theoretical bounds here that are useful in practice?

1 Description of work

The work I carried out during the 3 months at Yandex is still continuing and falls largely under a non disclosure agreement.[1] But in broad terms, my work consisted of the following. We investigated OL2R algorithms, specifically bandit algorithms, in the setting of queries related to current events. For such queries, there usually exist extremely new documents that did not yet attract any clicks or other user interaction features. Since those user interaction features are typically strong signals for ranking algorithms, those algorithms have a hard time pushing these extremely new documents into the top of a ranking. This, in turn, causes these new documents to never attract any clicks, as typically only the top ranked documents attract clicks. For this reason, it is crucial to perform a certain degree of *exploration*; place those very new documents at the top of a ranking even though there is no strong evidence (yet) that these documents are actually relevant documents. Placing them at the top of rankings will give users the opportunity to interact with these documents. These interactions, or their absence, will inform the engine about the relevance of the document for future issues of the same query. We investigated log data related to these issues and investigated the size and degree to which this occurs. Furthermore, we have experimented with algorithms performing this form of OL2R with real users and we are currently in the process of the analyzing results of these experiments.

2 Description of results

Concrete results of the exchange visit are the following:

- a good understanding of the inner workings of a large scale commercial search engine;
- practical experience in working with a large development team;
- insight in actual user interaction data;
- an implementation of an OL2R algorithm and an experiment with real users; and
- the beginning of a paper describing our experiments.

[1] We are in the process of writing a paper that will naturally be publicly available though.

3 Future collaboration with host institution

Currently the host and I are still collaborating in order to finish up the work done while I was at Yandex. Future collaboration is ensured through already existing ties between the University of Amsterdam, through a shared PhD programme. The exchange visit strengthened these ties and laid a strong foundation for future collaborations. Since my research thrives with the presence of real users, and given the abundance of real users present at Yandex it is likely that there will be further collaborations in the future.

4 Projected publications resulting from the grant

We plan on consolidating the work carried out during my visit. We are currently analyzing gathered data. We are targeting the The Eighth ACM International WSDM Conference with a full paper for which the applicant and the host institution intend to collaborate.

5 References

Chuklin, A., Schuth, A., Hofmann, K., Serdyukov, P., and de Rijke, M. (2013). Evaluating aggregated search using interleaving. In *CIKM '13*. ACM Press.

Hofmann, K., Schuth, A., Bellogin, A., and de Rijke, M. (2014). Effects of Position Bias on Click-Based Recommender Evaluation. In *ECIR '14*.

Hofmann, K., Schuth, A., Whiteson, S., and de Rijke, M. (2013). Reusing Historical Interaction Data for Faster Online Learning to Rank for IR. In *WSDM '13*.

Radlinski, F., Kurup, M., and Joachims, T. (2008). How does clickthrough data reflect retrieval quality? In *CIKM '08*. ACM Press.

Schuth, A., Hofmann, K., Whiteson, S., and de Rijke, M. (2013). Lerot: an Online Learning to Rank Framework. In *LivingLab '13*. ACM Press.

Schuth, A., Sietsma, F., Whiteson, S., and de Rijke, M. (2014a). In *Submitted*.

Schuth, A., Sietsma, F., Whiteson, S., and de Rijke, M. (2014b). Optimizing Base Rankers Using Clicks: A Case Study using BM25. In *ECIR '14*.

Yue, Y. and Joachims, T. (2009). Interactively optimizing information retrieval systems as a dueling bandits problem. In *ICML '09*.