Using Intent Information to Model User Behavior in Diversified Search

Aleksandr Chuklin[1,2], Pavel Serdyukov[1], and Maarten de Rijke[2]

[1] Yandex, Moscow, Russia
{chuklin,pavser}@yandex-team.ru,
[2] ISLA, University of Amsterdam
{A.Chuklin,deRijke}@uva.nl

Abstract. A result page of a modern commercial search engine often contains documents of different types targeted to satisfy different user intents (news, blogs, multimedia). When evaluating system performance and making design decisions we need to better understand user behavior on such result pages. To address this problem various click models have previously been proposed. In this paper we focus on result pages containing fresh results and propose a way to model user intent distribution and bias due to different document presentation types. To the best of our knowledge this is the first work that successfully uses intent and layout information to improve existing click models.

1 Introduction

The idea of search result diversification appeared several years ago in the work by Radlinski and Dumais [23]. Since then all major commercial search engines addressed the problem of ambiguous queries either by the technique called *federated* / *vertical* search (see, e.g., [2]) or by making result diversification a part of the ranking process [1, 25]. In this work we focus on one particular vertical: *fresh* results, i.e., recently published webpages (news, blogs, etc.). Fig. 1 shows part of a search engine result page (SERP) in which fresh results are mixed with ordinary results in response to the query "Chinese islands". We say that every document has a *presentation type*, in our example "fresh" (the first two documents in the figure) or "web" (the third, ordinary search result item). We will further refer to the list of presentation types for the current result page as a *layout*. We assume that each query has a number of *categories* or *intents* associated with it. In our case these will be "fresh" and "web".

The main problem that we address in this paper is the problem of modeling user behavior in the presence of vertical results. In order to better understand user behavior in a multi-intent environment we propose to exploit intent and layout information in a click model so as to improve its performance. Unlike previous click models our proposed model uses additional information that is already available to search engines. We assume that the system already knows the probability distribution of intents / categories corresponding to the query. This is a typical setup for the TREC diversity track [9] as well as for commercial search systems. We also know the presentation type of each document. We argue that this presentation may lead to some sort of bias in user behavior and taking it into account may improve the click model's performance.

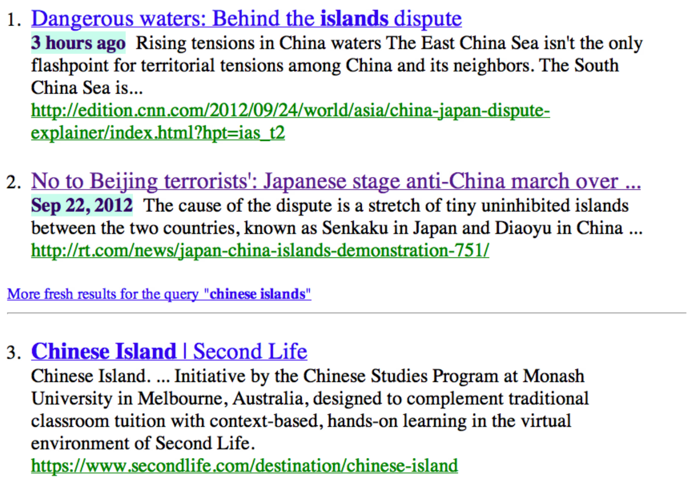


Fig. 1: Group of fresh results at the top followed by an ordinary search result item.

The main contribution of the paper is a novel framework of *intent-aware* (IA) *click models* that can be used to better understand various aspects of user behavior and document relevance in a diversified search setup:

- We propose to use presentation types of the documents on a SERP and prior knowledge about user intent distribution. Moreover, we propose a dynamic adaptation of this distribution (using previous clicks) when predicting next click(s) (see Eq. 7).
- We consider individual relevance values for different intents. It helps us improve model performance and enables new possible applications of click models.

2 Click Models

In order to show how layout and intent information can be used to better understand user behavior we propose modifications to commonly used click models. The information added through our modifications helps us improve click model performance.

2.1 Background

Click data has always been an important source of information for web search engines. It is an *implicit* signal because we do not always understand how user behavior correlates with user satisfaction: user's clicks are biased. Following Joachims et al. [18], who conducted eye-tracking experiments, there was a series of papers that model user behavior using probabilistic graphical models (see [19] for a general introduction). The most influential works in this area include the UBM model by Dupret and Piwowarski [13], the Cascade Model by Craswell et al. [11] and the DBN model by Chapelle and Zhang [7].

A *click model* can be described as follows. When a user submits a query q to a search engine she gets back 10 results: $u_1, \ldots, u_{10}$. Given a query q we denote a *session* to be a set of events experienced by the user since issuing the query until abandoning the result page or issuing another query. Note that one session corresponds to exactly one query. The minimal set of random variables used in all models to describe user behavior are: *examination* of the k-th document (E_k) and *click* on the k-th document (C_k):

- E_k indicates whether the user looked at the document at rank k (hidden variables).
- C_k indicates whether the user clicked on the k-th document (observed variables).

In order to define a click model we need to denote dependencies between these variables. For example, for the UBM model we define

$$P(E_k = 1 \mid C_1, \dots, C_{k-1}) = \gamma_{kd} \tag{1}$$

$$E_k = 0 \Rightarrow C_k = 0 \tag{2}$$

$$P(C_k = 1 \mid E_k = 1) = a_{u_k}, \tag{3}$$

where γ_{kd} is a function of two integer parameters: the current position k and the distance to the rank of previous click $d = k - PrevClick = k - \max\{j \mid 0 \leq j < k \;\&\; C_j = 1\}$ (we assume $C_0 = 1$). Furthermore, a_{u_k} is a variable responsible for the attractiveness of the document u_k for the query q. If we know the a and γ parameters, we can predict click events. The better we predict clicks the better the click model is.

2.2 Proposed modifications

We propose a modification to existing click models that exploits information about user intent and the result page layout. As a basic model to modify we use the UBM click model by Dupret and Piwowarski [13]. However, our extensions can equally well be applied to other click models. Unlike [8], we focus on HTML results that look very similar to the standard 10 blue links. We do not know beforehand that the user notices any differences between special (vertical) results and ordinary ones.

We add one hidden variable I and a set of observed variables $\{G_k\}$ to the two sets of variables $\{E_k\}$ and $\{C_k\}$ commonly used in click models:

- $I = i$ indicates that the user performing the session has *intent* i, i.e., relevance with respect to the category i is much more important for the user.
- $G_k = l$ indicates that the result at position k uses a presentation specific to the results with dominating intent l. For example, for the result page shown in Fig. 1 we have $G_1 = \mathit{fresh}$, $G_2 = \mathit{fresh}$, $G_3 = web$. We will further refer to a list of presentation types $\{G_1, \dots, G_{10}\}$ for a current session as a *layout*.

A typical user scenario can be described as follows. First, the user looks at the whole result page and decides whether to examine the k-th document or not. We assume that the examination probability $P(E_k)$ does not depend on the document itself, but depends on the user intent, her previous interaction with other results, the document rank k and the SERP layout. If she decides to examine the document (if $E_k = 1$) we assume that she is *focused* on that particular document. It implies that the probability of the click $P(C_k = 1|E_k = 1)$ depends only on the user intent I and the document relevance / attractiveness of the current document, but neither on the layout nor on the document position k. After clicking (or not clicking) the document the user moves to another document following the same “examine-then-click” scenario.

In this paper we only allow dependencies between E_k and G_k in order to simplify inference, but one can also consider additional dependency links.[3] As an example, using our proposed addition, one can build an intent-aware version of the UBM model in the following manner (cf. (1)–(3)):

$$P(E_k = 1 \mid G_k = b, I = i, C_1, \ldots, C_{k-1}) = \gamma_{kd}(b, i) \tag{4}$$

$$E_k = 0 \Rightarrow C_k = 0 \tag{5}$$

$$P(C_k = 1 \mid E_k = 1, I = i) = a^i_{u_k} \tag{6}$$

where a and γ are to be inferred from clicks: $a^i_{u_k}$ is the attractiveness of the document u_k for the intent i and $\gamma_{kd}(b, i)$ is the probability of examination given the distance to the previous click d, current intent $I = i$ and current presentation type $G_k = b$. The model is shown in Fig. 2. Below, we refer to (4)–(6) using *UBM-IA*.

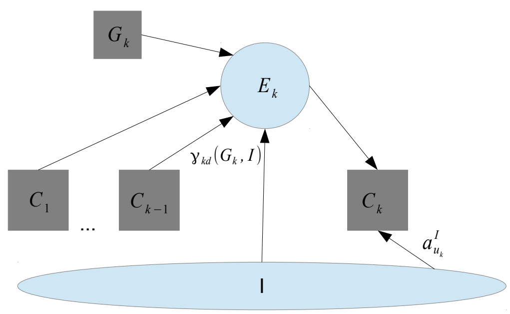


Fig. 2: The graphical model for UBM-IA. Gray squares correspond to observed variables, blue circles — hidden variables. Arrows show dependency links.

The aim of our work is not to study how to find intents corresponding to the query. Instead, given that we know the query intent spectrum, we aim to investigate the effect of this distribution on the users' click-through behavior. So we assume that for each session we have a prior distribution of the intents $P(I = i)$.[4] Importantly, unlike Hu et al. [17] we do not assume that our intent distribution is fixed for the session. When predicting the next click, we modify the intent distribution $P(I)$ using Bayes' rule:

$$P(C_k|C_1, \ldots, C_{k-1}) = \sum_I \underbrace{P(C_k|C_1, \ldots, C_{k-1}, I)}_{\text{probabiity from single intent model}} \cdot \underbrace{P(I|C_1, \ldots, C_{k-1})}_{\text{posterior intent distribution}} \tag{7}$$

Dupret and Piwowarski [13] find that the single browsing model outperforms a mixture of browsing models when inferring intent distribution from clicks. We show that using layout information and prior knowledge of intent distribution, we can significantly outperform the single browsing model.

3 Experimental Setup

The main research questions that our experiments are meant to answer are:

[3] For example, we can include the presentation type of a previous click(s) as it may indicate a bias towards documents of particular type. See our discussion in Section 5 for more details.

[4] In the current work we used a proprietary machine-learned algorithm to get this value.

- How do intent and layout information help in building click models? How does the performance change when we use only one type of information or both of them?
- How does the best variation of our model compare to other existing click models?

These questions are further discussed in Sections 4.1 and 4.2.

In order to test our ideas and answer our research questions, we collected a click log of the Yandex search engine and then used the Expectation-Maximization (EM) algorithm to infer model parameters; this algorithm is described in Appendix A. For our main experiment we used a sample of sessions with fresh results from a period of 30 days in July 2012. We discarded sessions with no clicks and did not take into account clicks on positions lower than ten. Fresh results are also counted and might appear at any position. We had 14,969,116 sessions with 2,978,309 different queries.

In order to verify the stability of our results we split the data into 30 subsets $\{b_t\}_{t=1}^{30}$ corresponding to successive days. We then used b_{2j-1} to train the model and b_{2j} to test it. So we measured how well the model can predict future clicks. We repeated the measurements for all $j \in \{1, \ldots, 15\}$ to verify significance of our results. We also experimented with the whole data (split into train and test sets) and observed almost identical results. To evaluate a model on a test set we used a standard *perplexity* metric: for each position k we calculated

$$p_k = 2^{-\frac{1}{N}\sum_{j=1}^{N}\left(C_k^j \log_2 q_k^j + (1-C_k^j)\log_2(1-q_k^j)\right)},$$

where C_k^j is a binary value corresponding to an observed click on the k-th position in the j-th session, q_k^j is the predicted probability of a click on the k-th position in the j-th session given the observed previous clicks. Perplexity measures how "surprised" the model is upon observing the click. The higher its value, the worst the model. Perplexity of a simple baseline model (predicting each click with probability 0.5) equals 2, perplexity of a perfect model is 1.

We also report an average perplexity value $AvgPerp = \frac{1}{10}\sum_{k=1}^{10} p_k$. To compute the perplexity gain of model B over model A we used a formula $\frac{p_A - p_B}{p_A - 1}$, which is a standard way to compare perplexity values.

4 Results

As a starting point we implemented the classical DBN and UBM models and tested them on our data. We found that the UBM model performs much better than DBN, consistently giving around 18% gain in perplexity over DBN. So we decided to use UBM as our baseline and we report our improvements compared to this model.

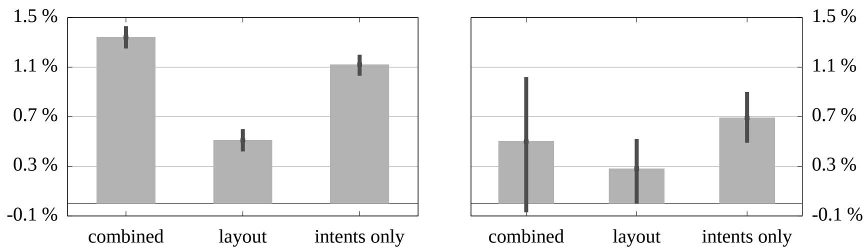
4.1 Layout and intent information

The combined contribution of layout and intent. We start by comparing our UBM-IA model (4)–(6) to the original UBM model and then consider the individual contributions of intent and layout information.

The main results are summarized in Table 1. In this table we report the average value of perplexity gain for the 15 subset pairs (b_t, b_{t+1}) described in the previous section. We also report confidence intervals calculated using the bootstrap method (see e.g., [14]). We can see that our improvements are statistically significant.

Table 1: Average perplexity gain for the combined UBM-IA model.

Model	Average Perplexity Gain	Confidence Interval (Bootstrap)
UBM-IA vs. UBM	1.34 %	[1.25%, 1.43%]



(a) Fresh results with special snippets (b) Fresh results with ordinary snippets

Fig. 3: Perplexity gains for *layout* and *intent* models compared to UBM.

Layout and intent in isolation. When we take a look at the modifications implemented on top of the UBM model, (4)–(6), we can see that they are actually a combination of two ideas: information about layout $\{G_k\}$ and information about user intents I. We can then test these ideas separately and see what their contribution is. We call the resulting click models *UBM-layout* and *UBM-intents*; they are defined using (5), (9), (10) and (5), (6), (8), respectively:

$$P(E_k = 1 \mid I = i, C_1, \ldots, C_{k-1}) = \gamma_{kd}(i) \tag{8}$$

$$P(E_k = 1 \mid G_k = b, C_1, \ldots, C_{k-1}) = \gamma_{kd}(b, i) \tag{9}$$

$$P(C_k = 1 \mid E_k = 1) = a_{u_k} \tag{10}$$

The results, in terms of perplexity, of comparing *UBM-IA*, *UBM-layout* and *UBM-intents* against UBM are summarized in Fig. 3(a). We can see that both individual models give some improvement while the best results are achieved using the combined model *UBM-IA*. Using the bootstrap method we confirm that the observed differences are statistically significant; the confidence intervals are shown as vertical bars.

The importance of layout information. How much of the positive effects observed in Fig. 3(a) is due to layout information, that is, to the fact that fresh results are singled out and clearly presented as such? In order to answer this question we performed the following user experiment. A small part of all Yandex users were presented with fresh results that looked just like ordinary documents while placed on the same positions. In other words, despite the fact that the search engine knows the presentation type G_k of every document, these users could not see it. We hypothesize that the usage of layout information will be less reliable in this situation because users with *fresh* intent are less inclined to examine these documents. We collected the data for a period of 12 days in September 2012 and evaluated the same three click models (*UBM-IA*, *UBM-layout* and *UBM-intents*) on this data.

The results, again in terms of perplexity gain, are shown in Fig. 3(b). Because we have much less data (121,431 sessions corresponding to 42,049 unique queries) our bootstrap confidence intervals are wide, wider than in Fig. 3(a). From the plot we see that only including layout information does not help, and that the best model in this situation is *UBM-intents*, which affirms our hypothesis.

Gain per rank. The results so far report on perplexity gains over the complete SERP. We now examine the perplexity gains per individual ranking position to analyze our click models in more detail. Fig. 4 shows the results for all three models: *UBM-IA* (combined), *UBM-layout* and *UBM-intent*. One can see that it is difficult to make an improvement for the first document because the models do not differ much for the first position: users usually examine the first document despite its presentation type and other factors, and therefore click probability is motivated only by the (perceived) relevance of the document. Clicks on the last two positions are not motivated by user intent or page layout: this information even leads to a decrease in perplexity for the *UBM-intent* and *UBM-IA* click models. However, *UBM-layout* is robust to such errors: it always gives an improvement even if it is mostly smaller than that of other models.

There is another interesting observation to be made. Intent information matters for positions 2–6, while layout information matters for positions 2–10, and it is more important than intent for positions 6–10. This change can be explained by the fact that for most of the users only the first 5–6 documents can be viewed without scrolling.

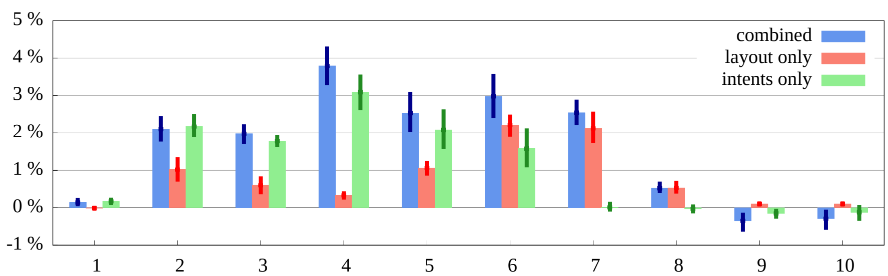


Fig. 4: Perplexity gains for different ranking positions compared to UBM model.

4.2 Other models

We also implemented the DBN [7], DCM [15] and RBP [21] click models. Since these models all performed significantly worse than UBM on our data, they also performed worse than our *UBM-IA* click model.

As we mentioned previously, Chen et al. [8] also addressed the problem of verticals by a click model. We can consider their click model as a state-of-the-art click model for diversified search. While the main focus of that work was on visually appealing verticals (containing images or video) we can assume that our fresh results are more or less similar to their *News* vertical. We then used the best performing click model for that

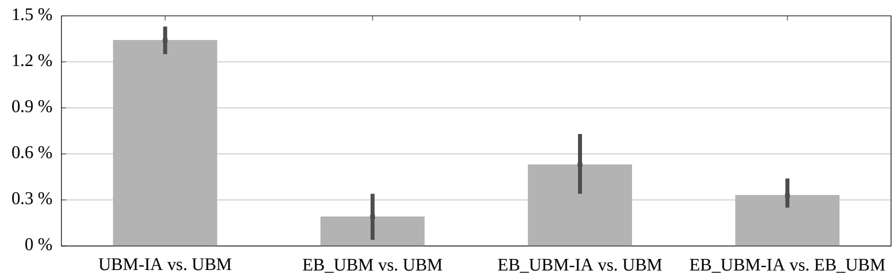


Fig. 5: Perplexity gains for different models.

vertical called "Exploration Bias Model" that was based upon UBM. Here, we refer to it as EB_UBM.

Our EB_UBM-IA click model extends the EB_UBM click model in the same way as it does for UBM (as specified in the Appendix). We compare UBM, UBM-IA, EB_UBM and EB_UBM-IA on our data set. A short summary is reported in Fig. 5. We see that our UBM-IA click model gives a bigger improvement over the original UBM model than EB_UBM. We can also see that if we combine our ideas of layout and intent with Chen et al.'s idea of "exploration bias" (yielding the EB_UBM-IA click model), we observe a gain over EB_UBM but not as big as for UBM-IA. It means that we *can* combine these approaches, but we should *not* do it, because they seem to interfere.

5 Discussion and Related Work

The problem we addressed in this paper is closely related to the search result diversification described in [23]. Historically, there are two different approaches to this problem. We can call them the *intent* and *vertical* approaches.

The *intent* hypothesis assumes that each document has separate relevances for different user intents.[5] Following this hypothesis, a family of intent-aware (IA) metrics arose [1] as well as other metrics addressing the problem of diversity: ERR-IA [6], α-$NDCG$ [10], $D\sharp$-$NDCG$ [24]. ERR-IA has become a standard ranking function for the TREC diversity track now (see [9] for more details). There were also attempts to use intent information in click models. In the original publication on UBM by Dupret and Piwowarski [13], a so-called *mixture model* was studied. Instead of using prior knowledge of intent distribution they learned such information from clicks and were not able to report any improvements compared to a single browsing model. In a later publication, Hu et al. [17] proposed to use a constant relevance discount factor for each session to model intent bias. While their approach is valid for building a click model, the variable they used to implicitly model intent bias does not correspond to the commonly used notion of *intents* or *categories* to which we adhere in the current work.

Vertical or *federated* search is an approach adopted by many modern search engines. Following this approach an incoming query is sent to several specialized search engines, called *verticals* (e.g., images, video, news). If the results from some vertical are suitable for the query, they are placed in a grouped manner somewhere on a search engine result page. Usually there are three or four insert positions ("slots") where vertical results

[5] Frequently referred to as *categories*, *topics* or *nuggets* [10].

can be placed, so presentation is fixed. The two problems that are usually studied in previous works are vertical selection and vertical ranking:

- *Vertical selection*: determine which verticals are relevant to the given query.
- *Vertical ranking*: decide which vertical block should be placed higher than others.

There are several papers following the vertical approach that address the problems above (e.g., [2, 3, 5]) as well as the problem of result evaluation [4, 22]. Recent work by Chen et al. [8] on click models also follows the vertical approach.

Since the publication of the original DBN and UBM click models, there have been many papers that address different types of bias in click models. DBN itself was based on a *Cascade Model* [11]. The main idea of the cascade hypothesis was that the user examines documents one by one and the examination probability of a document depends on the relevance of the document above it. Another cascade-based model is the *Dependent Click Model* [15], which was later followed by the *Click Chain Model* [16]. Liu et al. [20] use the same UBM model but proposed a faster algorithm for parameter inference. Zhang et al. [26] go beyond a single query by modeling user tasks in a *Task-centric Click Model*.

In [2], where the problem of vertical selection is studied in detail, there is a list of commonly used verticals such as *news*, *images*, *video*, *TV*, *sports*, *maps*, *finance*, etc. Most of these contain images or interactive tools like video or maps. On the one hand, the fact that we focused on fresh results can be viewed as a limitation of our work. On the other hand, there are many user intents that can be (and should be) covered by more or less textual results: *official pages*, *forums*, *blogs*, *reviews*, etc.

An early study by Dumais et al. [12] suggests that users tend to prefer grouped results as they are easier to investigate. It took less time for participants of their experiment to complete search tasks using a grouped interface. However, if we optimize a diversity metric (e.g., ERR-IA) we will end up with a blended result page where results are not necessarily grouped. To address this problem we ran an online AB-testing experiment where some users were presented with fresh results grouped while other users always saw fresh results mixed with ordinary web results. We found that fresh results got 5% fewer clicks when they are mixed with other results while the total number of clicks and abandonments remained unchanged.[6] This suggests that if we want to optimize traffic on fresh results (e.g., if news content providers share some revenue with the web search company) we need to consider the fact that user behavior depends on how we organize vertical results. One can extend our intent-aware click model to handle these types of layout changes by introducing additional dependencies between the examination probability E_k and the page layout $\{G_k\}_{k=1}^{10}$. For example, for our *UBM-IA* click model we can add dependency on the number of vertical groups or presentation type of the previous document G_{k-1} to the γ_{kd} function (see (4)).

6 Conclusion

The main contribution of our work is a framework of intent-aware click models, which incorporates both layout and intent information. Our intent-aware modification can be

[6] The difference is significant at level $\alpha = 0.001$ when using two-tailed Mann-Whitney U test.

applied to any click model to improve its perplexity. One interesting feature of an intent aware click model is that it allows us to infer separate relevances for different intents from clicks. These relevances can be further used as features for specific vertical ranking formulas. Another important property of intent-aware additions to click models is that by analyzing examination probabilities (e.g., γ_{rd} in the case of UBM) we can see how user patience depends on his/her intent and SERP layout. Put differently, it allows us to use a click model as an ad-hoc analytic tool.

As to future work, we see a number of directions, especially concerning specific verticals in order to check that our method is also applicable to other verticals/intents. We mention two examples. First, the mobile arena provides interesting research opportunities. We performed a set of preliminary experiments using mobile applications as a vertical: a result item from this vertical consists of a text snippet with a small thumbnail, price and application rating. These documents are more visually appealing than fresh results but still look similar to web results (unlike video or images). The data was collected during several days in September 2012 and consisted of 34,917 sessions and 11,595 unique queries. We found that both *UBM-IA* and EB_UBM-IA gives an improvement of about 9% perplexity over UBM, while EB_UBM without our modifications only gives a 0.15% improvement. It would be interesting to perform a full-scale study of the model performance for different verticals as a future work.

Sometimes, intents are very unique, like for instance for the query "jaguar" there are at least two intents: finding information about *cars* and finding information about *animals*. It is very unlikely that a search engine has a special vertical for these intents. However, we believe that knowledge of the user's intent can still be used in order to better understand his/her behavior. Applying our ideas to these minor intents is an interesting direction for future work.

Acknowledgments. This research was partially supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements nr 258191 (PROMISE Network of Excellence) and 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.-814, 612.061.815, 640.004.802, 727.011.005, 612.001.116, HOR-11-10, the Center for Creation, Content and Technology (CCCT), the Hyperlocal Service Platform project funded by the Service Innovation & ICT program, the WAHSP and BILAND projects funded by the CLARIN-nl program, the Dutch national program COMMIT, by the ESF Research Network Program ELIAS, and Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW).

7 References

[1] Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: WSDM. p. 5. ACM (2009)
[2] Arguello, J., Diaz, F., Callan, J., Crespo, J.: Sources of evidence for vertical selection. In: SIGIR. pp. 315–322. ACM (2009)
[3] Arguello, J., Diaz, F., Callan, J.: Learning to Aggregate Vertical Results into Web Search Results. In: CIKM. ACM (2011)

[4] Arguello, J., Diaz, F., Callan, J., Carterette, B.: A methodology for evaluating aggregated search results. In: ECIR. Springer (2011)
[5] Arguello, J., Diaz, F., Paiement, J.: Vertical selection in the presence of unlabeled verticals. In: SIGIR. ACM (2010)
[6] Chapelle, O., Metzler, D., Zhang, Y., Grinspan, P.: Expected reciprocal rank for graded relevance. In: CIKM. p. 621. ACM (2009)
[7] Chapelle, O., Zhang, Y.: A dynamic bayesian network click model for web search ranking. In: WWW. ACM (2009)
[8] Chen, D., Chen, W., Wang, H.: Beyond ten blue links: enabling user click modeling in federated web search. In: WSDM. ACM (2012)
[9] Clarke, C.L.A., Craswell, N., Soboroff, I.: A comparative analysis of cascade measures for novelty and diversity. In: WSDM. pp. 75–84. ACM (2011)
[10] Clarke, C., Kolla, M., Cormack, G.: Novelty and diversity in information retrieval evaluation. In: SIGIR. ACM (2008)
[11] Craswell, N., Zoeter, O., Taylor, M., Ramsey, B.: An experimental comparison of click position-bias models. In: WSDM. p. 87. ACM (2008)
[12] Dumais, S., Cutrell, E., Chen, H.: Optimizing search by showing results in context. In: CHI. ACM (2001)
[13] Dupret, G., Piwowarski, B.: A user browsing model to predict search engine click data from past observations. In: SIGIR. pp. 331–338. SIGIR '08, ACM (2008)
[14] Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap (Chapman & Hall/CRC Monographs on Statistics & Applied Probability). Chapman and Hall/CRC, 1 edn. (May 1994)
[15] Guo, F., Liu, C., Wang, Y.: Efficient multiple-click models in web search. In: WSDM. ACM (2009)
[16] Guo, F., Liu, C., Kannan, A., Minka, T., Taylor, M., Wang, Y.M., Faloutsos, C.: Click chain model in web search. In: WWW. p. 11. ACM (2009)
[17] Hu, B., Zhang, Y., Chen, W., Wang, G., Yang, Q.: Characterizing search intent diversity into click models. In: WWW. p. 17. ACM (2011)
[18] Joachims, T., Granka, L., Pan, B., Hembrooke, H., Gay, G.: Accurately interpreting clickthrough data as implicit feedback. In: SIGIR. p. 154. ACM (2005)
[19] Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (Aug 2009)
[20] Liu, C., Guo, F., Faloutsos, C.: BBM. In: KDD. p. 537. ACM (Jun 2009)
[21] Moffat, A., Zobel, J.: Rank-biased precision for measurement of retrieval effectiveness. ACM Transactions on Information Systems 27(1), 1–27 (Dec 2008)
[22] Ponnuswami, A.K., Pattabiraman, K., Wu, Q., Gilad-Bachrach, R., Kanungo, T.: On composition of a federated web search result page: using online users to provide pairwise preference for heterogeneous verticals. In: WSDM. ACM (2011)
[23] Radlinski, F., Dumais, S.: Improving personalized web search using result diversification. In: SIGIR. ACM (2006)
[24] Sakai, T., Song, R.: Evaluating diversified search results using per-intent graded relevance. In: SIGIR. ACM (2011)
[25] Styskin, A., Romanenko, F., Vorobyev, F., Serdyukov, P.: Recency ranking by diversification of result set. In: CIKM. pp. 1949–1952. ACM (2011)
[26] Zhang, Y., Chen, W., Wang, D., Yang, Q.: User-click modeling for understanding and predicting search-behavior. In: KDD. ACM (2011)

A Appendix

We describe an Expectation-Maximization algorithm for the UBM-IA click model. Algorithms for the other models considered in the paper can be derived in a similar manner. In order to simplify our calculations we use two sets of hidden variables: $\{E_k\}$ (the user examined the k-th document) and $\{A_k\}$ (the user was attracted by the k-th document). The advantage of using auxiliary variables is that for every parameter we need to infer we have a corresponding hidden variable. By using these variables we can rewrite the main UBM equations ((4)–(6)) as follows:

$$P(E_k = 1 \mid G_k = b, I = i, C_1, \ldots, C_{k-1}) = \gamma_{kd}(b, i)$$
$$P(A_k = 1 \mid I = i) = a_u^i$$
$$E_k = 1, A_k = 1 \Leftrightarrow C_k = 1$$

Suppose that we have N sessions and a record of URLs shown, their visual representations G_k and click positions. Let us denote the vectors of observed variables as C^j and G^j and the vectors of hidden variables as E^j and A^j. We also use a vector d^j representing 10 documents shown during the j-th session (we cut off sessions that have clicks on further pages). Each vector has length 10, e.g., C_k^j is a binary variable denoting whether the k-th document was clicked in the j-th session. We use I^j to denote a hidden variable representing the intent for this session.

M-step. At the M-step we estimate the vector of parameters θ from the previous estimation θ^t:

$$\theta^{t+1} = \arg\max_\theta \sum_y P(Y = y \mid X, \theta^t) \log P(X, Y \mid \theta) \; , \tag{11}$$

where X and Y denote the sets of observed and hidden variables respectively. In our case:

$$a_u^i = \arg\max_a \sum_{j=1}^{N} \sum_{k=1}^{10} I(d_k^j = u)$$
$$(q_{A_k}(0, i) \log(1 - a) + q_{A_k}(1, i) \log a) + \log P(a)$$
$$\gamma_{kd}(b, i) = \arg\max_\gamma \sum_{j=1}^{N} \sum_{k=1}^{10} I(d_k^j = u, G_k^j = b, PrevClick = d)$$
$$(q_{E_k}(0, i) \log(1 - \gamma) + q_{E_k}(1, i) \log \gamma) + \log P(\gamma)$$

where $P(a)$, $P(\gamma)$ are beta priors and q_{A_k}, q_{E_k} are calculated during the E-step.

E-step. Let us first define the probabilities we need to compute:[7]

$$q_{A_k}(a, i) = P(A_k = a, I = i \mid C, G) \tag{12}$$
$$q_{E_k}(e, i) = P(E_k = e, I = i \mid C, G) \tag{13}$$

We can transform (12) and (13) using Bayes' rule. E.g. for A_k we have:

$$P(A_k, I \mid C, G) = P(A_k \mid I, C, G) \cdot P(I \mid C, G)$$

The probability $P(I \mid C, G)$ can be calculated as follows:

$$P(I \mid C, G) = \frac{P(C \mid I, G) P(I)}{\sum_{i'} P(C \mid I = i', G) P(I = i')}, \tag{14}$$

where $P(I)$ is a prior distribution of intents for a query (assumed to be known). Now, if $C_k = 0$:

$$P(A_k = 1 \mid I = i, C, G) = \frac{a_u^i (1 - \gamma_{kd}(b, i))}{1 - a_u^i \gamma_{kd}(b, i)},$$
$$P(E_k = 1 \mid I = i, C, G) = \frac{\gamma_{kd}(b, i)(1 - a_u^i)}{1 - a_u^i \gamma_{kd}(b, i)}.$$

If $C_k = 1$ then $P(A_k = 1 \mid I = i, C, G) = 1$ and $P(E_k = 1 \mid I = i, C, G) = 1$. By combining these equations with (14) we complete the E-step.

[7] We omit the superscript j here for convenience.