Click models evaluations
(IR2 project)

Luka Stout
10616713
and Finde Xumara
10690832

No Institute Given

1 Introduction

Modeling user behavior on a search engine result page is important for understanding the users and supporting simulation experiments. As result pages become more complex, click models have to evolve as well in order to capture additional aspects of user behavior in response to new forms of result presentation. In recent years many models have been proposed that are aimed at predicting the behaviour of web search users.

In this report, we will implement and evaluate different click models using multiple evaluation metrics. These click models include the Click-Through Rate click model (CTR), Position-Based click Model (PBM), Cascade click model (CM) [9], Dependent Click Model (DCM) [6], Dynamic Bayesian Network click model (DBN) [1], User Browsing Model (UBM) [4], Click Chain Model (CCM) [5] and Task-centric Click Model (TCM) [12]. The evaluation metrics include likelihood, perplexity, click through rate prediction, relevance prediction, ranking performance and computation time. By doing this experiment we can know the performance of each different click model and this information can be used as a benchmark of the next click model proposal.

This report is organized as follows. In Section 2 we will describe the different click models and evaluation algorithms used in our experiments. The experiments and data source information will be covered in Section 3, followed by an analysis of these experiments in Section 5.

2 Methodology

The different models we compare UBM, DCM, DBN and TCM on the Yandex dataset [10]. In this section, we briefly describe their main characteristics and differences. We have implemented TCM ourselves, the other algorithms were taken from PyClick [11].

2.1 Click-through rate click model

The most simple click model, abbreviated as CTR, that actually tries to predict relevance for a query-documenent pair. The relevance is the only parameter in this model. The relevance in this model is simple defined as $a_u = \frac{\#\, of\, clicks}{\#\, of\, times\, shown}$. We included this model as a simple baseline that the models should improve upon.

2.2 Position based model

This model builds upon the CTR model. It adds a `position bias` where documents in a higher position are examined more often. A document can only be clicked if it was examined. The values for this position bias are taken from [8] and are $[.68, .61, .48, .34, .28, .2, .11, .1, .08, .06]$.

2.3 Cascade model

Cascade model assumes that users abandon the query session after the first click and hence does not provide a complete picture of how multiple clicks arise in a query session and how to estimate document relevance from such data [9,**?**]. This model also assume that the user views search results from top to bottom, deciding whether to click each result before moving to the next. To observe a click, the user must have decided both to click and skip the ranks above.

2.4 Dependent Click Model

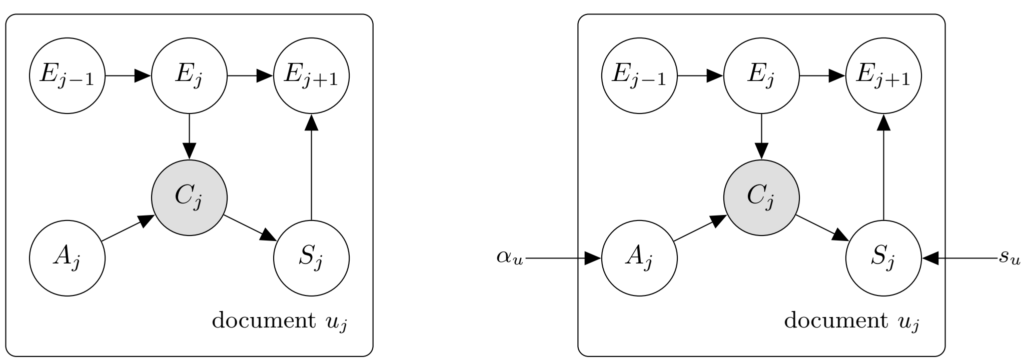
The dependent click model was first proposed by Guo et al. in [6]. In the paper they propose a new click model which can handle multiple clicks per query by introducing a position dependent parameter $lambda_j$ to reflect the chance that the user would like to see more results after a click at position j. A graphical representation of the model is presented in Figure1a

2.5 Dynamic Bayesion Network

The dynamic bayesian network click model is an extension to the traditional cascade model proposed by Chapelle and Zhang in [12]. For a given position j, in addition to observed variable C_j indicating whether there was a click or not at this position, the following latent variable are defined to model examination, perceived relevance and actual relevance, respectedly:

- E_j: Did the user *examine* the document?
- A_j: Was the user *attracted* by the document?
- S_j: Was the user *satisfied* by the clicked document?

They introduce a variable s_u for each document u which describes the relevance of the document for this query. When the user clicks on this document, there is a certain chance that the user will be satisfied. If the user is not satisfied, he continues to examine the next document with a probability γ and stops otherwise. The parameter γ is known as the 'perseverance'. A graphical representation of the model is presented in Figure 1b.



(a) DCM (b) DBN

Fig. 1: The graphical model of DCM and DBN.

2.6 User Browsing Model

In [4], Dupret and Piwowarski propose a new click model called the User Browsing Model (UBM). The main difference between UBM and other models is that UBM takes the distance into account from the current document u_j to the last clicked document $u_{j'}$ for determining the probability that the user continues browsing:

$$P(E_j = 1 \mid C_{j'} = 1, C_{j'+1} = 0, \dots, C_{j-1} = 0) = \gamma_{jj'}$$

The probability that a document at rank j is examined E_j therefore depends on all possible paths the user could have taken to arrive at this document:

$$P(E_j = 1) = \sum_{j'=1}^{j-1} \gamma_{jj'}$$

A graphical representation of the model is presented in Figure 2.

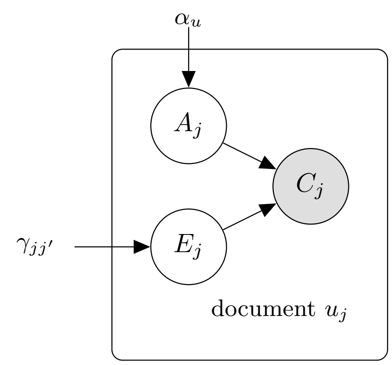


Fig. 2: The graphical model of UBM.

2.7 Click Chain Model

In [5], Fan Guo et al, proposed a bayesian based click model which was based on position bias assumption, named Click chain model (CCM). CCM shares the following assumptions with the cascade model and DCM: (1) users are homogeneous: their information needs are similar given the same query; (2) decoupled examination and click events: click probability is solely determined by the examination probability and the document relevance at a given position; (3) cascade examination: examination is in strictly sequential order with no breaks.

This model can be formalized with the following conditional probabilities in which R_j is the relevance variable of document u at position j, E_j is the examination variable, and αs form the set of user behavior parameters:

$$P(C_j = 1 | E_j = 0) = 0 \tag{1}$$
$$P(C_j = 1 | E_j = 1, R_j) = R_j \tag{2}$$
$$P(E_{j+1} = 1 | E_j = 0) = 0 \tag{3}$$
$$P(E_{j+1} = 1 | E_j = 1, C_j = 0) = \alpha_1 \tag{4}$$
$$P(E_{j+1} = 1 | E_j = 1, C_j = 1, R_j) = \alpha_2(1 - R_j) + \alpha_3 R_j \tag{5}$$
$$\tag{6}$$

A graphical representation of the model is presented in Figure 3.

2.8 Task-centric Click Model

The Task-centric Click Model (TCM) was first proposed by Zhang et al. in [12]. In the paper they propose a new click model which can handle multiple clicks of multiple queries in a task by introducing two new

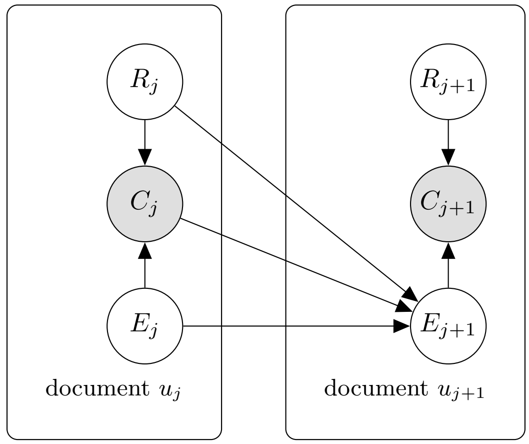


Fig. 3: The graphical model of CCM.

biases. The first bias indicates that users tend to express their information needs incrementally in a task, thus perform more clicks as their needs become clearer. The other bias indicates that users tend to click fresh documents that are not included in the results of previous queries. In their paper, they named the first assumption as `query bias`, and the second assumption as `duplicate bias`. A graphical representation of the state-of-the-art of the model is presented in Figure1a and the notations used in TCM are described in Table1.

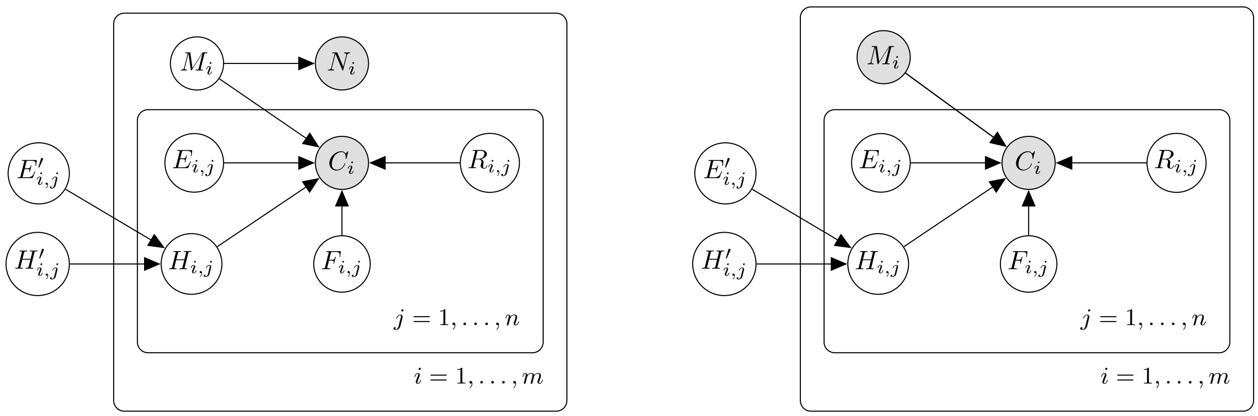
Symbol	Description
(i,j)	j-th ranking position in i-th query session.
M_i	Whether the i-th query matches the user's intent.
N_i	Whether the the user submits another query after i-th query session.
$E_{i,j}$	Examination of the document at (i,j).
$H_{i,j}$	Previous Examination of the document at (i,j).
$F_{i,j}$	Freshness of the document at (i,j).
$R_{i,j}$	Relevance of the document at (i,j).
$C_{i,j}$	Whether the the document at (i,j) is clicked.
(i',j')	Assume that d is the document at (i,j). i' is the latest query session where d has appeared in previous query sessions, and j' is the ranking position of this appearance.

Table 1: Notations used in TCM

This model can be formalized with the following conditional probabilities:

$$P(M_i = 1) = \alpha_1 \tag{7}$$
$$P(N_i|M_i = 1) = \alpha_2 \tag{8}$$
$$P(F_{i,j} = 1|M_{i,j} = 1) = \alpha_3 \tag{9}$$
$$P(E_{i,j} = 1) = \beta_j \tag{10}$$
$$P(R_{i,j} = 1) = r_d \tag{11}$$
$$M_i = 0 \Rightarrow N_i = 1 \tag{12}$$
$$H_{i,j} = 0 \Rightarrow F_{i,j} = 1 \tag{13}$$
$$H_{i,j} = 0 \Leftrightarrow H_{i',j'} = 0, E_{i',j'} = 0 \tag{14}$$
$$C_{i,j} = 1 \Leftrightarrow M_i = 1, E_{i,j} = 1, R_{i,j} = 1, F_{i,j} = 1 \tag{15}$$

In our implementation, we simplified TCM model by assuming that M_i is observed from the click log data, thus eq.8 can be removed. Our second assumptions is that $M_i, E_{i,j}, R_{i,j}$ and $F_{i,j}$ are independent. The detail calculation for updating EM parameters of the simplified TCM can be found in the appendices section. The graphical model of our TCM implementation is presented in Fig 4b.



(a) state-of-the-art TCM.

(b) simplified TCM

Fig. 4: The graphical model of state-of-the-art TCM and simplified TCM.

3 Evaluation Measures

To equally evaluate each click model's performance, we use evaluation metrics that are proposed along with the click models. The evaluation metrics used in this experiment are listed below:

3.1 Loglikelihood

1Loglikelihood is the default evaluation metric in machine learning. It says something about the likelihood of the data give the model. In Equation 16 the calculation of the loglikelihood of a click model and a set of sessions can be seen, where $\mathcal{LL}(S|\mathbf{M})$ is the loglikelihood of the sessions given the models parameters, S the set of sessions, $\mathbf{M}$ the model and its parameters, r is the rank of a particular document in a session and $c_r^{(s)}$ a indicator function that is 1 when the document at rank r in session s was clicked and 0 otherwise.

$$\mathcal{LL}(S|\mathbf{M}) = \frac{1}{|S|} \sum_{s \in S} \frac{1}{|s|} \sum_{r=1}^{|s|} \log P(C_r = c_r^{(s)} | M, C_{r-1}, C_{r-2} \ldots C_1) \tag{16}$$

3.2 Perplexity

Click perplexity is a widely used metric for evaluating click model accuracy. It measures how surprised a model is to see $c_r^{(s)}$ under the current parameters. Perplexity is calculated for every rank individually, as to see whether some models perform better on documents higher on the SERP page then documents ranked lower on the SERP. It is used as a evaluation metric in [12] and [4]. The calculation of perplexity can be seen in Equation 17.

$$\begin{aligned} Perplexity_r &= 2^{-\frac{1}{|S|} \sum_{s \in S} (c_r^{(s)} \log_2 p_r^{(s)} + (1 - c_r^{(s)}) \log_2 (1 - p_r^{(s)}))} \\ p_r^{(s)} &= P(C_r = 1 | \mathbf{M}) \end{aligned} \tag{17}$$

The perplexity of a data set is defined as the average of perplexities over all positions. Thus, a smaller perplexity value indicates a better consistency between the click model and the actual click data.

3.3 Click-Trough-Rate Prediction (CTR)

The purpose of click-through rates is to measure the ratio of clicks to impressions of an document. Generally the higher the CTR the higher chance of that document being clicked. The click-through rate of a document d is defined as:

$$CTR_d = \frac{1}{|S_d|} \sum_{s_d} c_{r_d}^{(s_d)} \tag{18}$$

where S_d is the set of sessions where document d appears. A way to use this as an evaluation measure is proposed in [1, p. 4]. In the same way we calculate the CTR prediction using the following protocol:

1. Retrieve all sessions related to a given query.
2. Consider an url that appears both in position 1 and some other positions.
3. Hold out as test sessions all the sessions in which that url appeared in position 1.
4. Train the model on the remaining sessions and predict the relevance.
5. Compute the test CTR in position 1 on the held-out sessions.
6. Compute an error between these two quantities.
7. Average the error on all such urls and queries, weighted by the number of test sessions.

The error measure we use is the Root-Mean-Square-Error (RMSE).

3.4 Relevance Prediction

Relevance prediction was used to evaluate performance of the DBN model [1, p. 6]. The accuracy of CTR prediction may not directly translate to relevance, especially when we were to evaluate the whole task instead of a single query. In this case, the CTR of a particular document is highly dependent on the user-model assumption. For example if a user tends to ignore a document that isn't fresh, CTR will be low even if the document is relevant. To measure relevance prediction we use a hand annotated set of relevances. We then use the Area Under the Curve (AUC) between the annotated relevances and the predicted relevances as an evaluation measure. We also measure the Pearson correlation between the two.

3.5 Predicted Relevance as a Ranking Feature

In this set of experiments we use the predicted relevance directly to rank urls, we use the model as a ranker. To evaluate the performance of a ranker we use the Normalized Discounted Cumulative Gain (NDCG) [7], for which we use a cutoff at five (NDCG@5). To calculate the NDCG@5 we only consider the documents for which we have annotated relevances. All these queries are then averaged to calculate the ranking performance of the click model. The algorithm can be seen below:

1. Retrieve all session that appear more than 10 times.
2. Filter out the sessions that don't appear in the editorial judgements.
3. Train the model on the sessions and predict relevance for the sessions.
4. Sort the urls w.r.t the predicted relevance given by the model.
5. Compute the NDCG@5.
6. Average over all sessions.

3.6 Computation Time

Historically in machine learning a big problem in creating accurate models was the amount of data that was available. However this is no longer the case, we are mostly restricted by the time that it takes to learn a model from the large amount of data that we currently have. So an important feature of a succesful click model is that it should be able to efficiently compute its parameters. Therefore, we also decided to look at the computation time it takes to train the click models.

TODO: Add some info about machine it is run on???

4 Experiments

In this section we report on the experimental evaluation and comparison of the click models in Section 2 on the evaluation metrics in Section 3 evaluated on the first 1 million query sessions from the Yandex Relevance Prediction contest of 2011 **??**. The rest of this section will elaborate on the experiments we have done and the results we found. We will also show how query frequency improves the performance of the models. Moreover, we will report on the evaluation measures in combination with click entropy.

4.1 Experimental Setup

The Yandex data set was used in the Yandex Relevance Prediction contest of 2011. In the experiments we have done we have used the first 1 million queries. We deliberatly have chosen to keep sessions that are without any clicks because removing them might hurt the performance of certain models, i.e. TCM because the freshness of a document can change with these sessions. These first 1 million sessions contain X distinct queries. These queries get divided in a set of training sessions used to train the click models and a set of test sessions used in the evaluation of the models, the number of sessions in these sets have a 3 to 1 ratio. The problem with this approach in combination with the TCM model is that this removes the guarantee that the complete task is used when training or evaluating the model. We expect that this will hurt the performance of the TCM model and that the actual performance might be higher.

To report on the performance with regard to query frequency we have split the data into 5 parts. The distribution of these parts can be seen in Table **??** Add table on query splitting. In the rest of this section with every evaluation measure we will also report on how this was influenced by the query frequency. The second factor that might influence performance that we have analyzed is the click entropy. The click entropy was also used to analyse queries in [3]. The formal definition is shown in Equation **??**.

$$ClickEntropy(q) = \sum_{d \in \mathcal{P}(q)} -P(d|q) \log_2 P(d|q) \tag{19}$$

Here $ClickEntropy(q)$ is the click entropy for query q. $\mathcal{P}(q)$ are documents clicked on when regarding query q. $P(d|q)$ is the percentage of clicks on document d among all clicks on q. If $c_{r_d}^{(q)}$ indicates a click on document d in query q then the click entropy is calculated as follows:

$$P(d|q) = \frac{\sum_p c_{r_d}^{(q)}}{\sum_{u \in \mathcal{P}(q)} c_{r_u}^{(q)}} \tag{20}$$

4.2 Results on Loglikelihood

In Figure **??** the results of the loglikelihood experiments can be seen.

4.3 Results on Perplexity

4.4 Results on CTR-Prediction

4.5 Results on Relevance Prediction

4.6 Results Predicted Relevance as a Ranking Feature

4.7 Results on Computation Time

In Table 2 one can see the results of the experiments.

5 Analysis

After running the experiments we were able to evaluate the different algorithms based on the ...

Model	Log-likelihood	Perplexity	Rel. Pred.	AUC Ranking	NDCG	CTR Pred.	Training time (sec)
UBM	**-0.288**	1.382	0.699	0.875	0.218	10387.7
TCM	-0.307	**1.374**	0.532	**0.977**	0.234	8145.77
SimpleDCM	-0.450	1.387	0.674	0.837	**0.210**	186.915
SimpleDBN	-0.448	1.386	0.668	0.740	**0.210**	**161.067**
DCM	-0.453	1.380	**0.706**	0.825	0.228	18332.5
DBN	-0.375	1.383	0.436	0.681	0.230	13204.2

Table 2: Results

6 Conclusions

In this paper we showed that ...

In our implementation, we did not ...

References

1. Olivier Chapelle and Ya Zhang. A Dynamic Bayesian Network Click Model for Web Search Ranking Categories and Subject Descriptors. *Www*, pages 1–10, 2009.
2. Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. *Proceedings of the 2008 . . .* , page 87, 2008.
3. Zhicheng Dou, Ruihua Song, Ji-Rong Wen, and Xiaojie Yuan. Evaluating the effectiveness of personalized web search. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2008.
4. Georges E Dupret and Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. *Sigir '08*, pages 331–338, 2008.
5. Fan Guo, Chao Liu, Anitha Kannan, Tom Minka, Michael Taylor, Yi-Min Min Wang, and Christos Faloutsos. Click chain model in web search. *Proceedings of the 18th international conference on World wide web - WWW '09*, page 11, 2009.
6. Fan Guo, Chao Liu, and Yi Min Ym Wang. Efficient multiple-click models in web search. *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 124–131, 2009.
7. K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.
8. Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, pages 154–161, New York, NY, USA, 2005. ACM.
9. David Kempe and Mohammad Mahdian. A cascade model for externalities in sponsored search. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5385 LNCS:585–596, 2008.
10. Yandex LLC. Task and datasets, n.d.
11. Ilya Markov. Click models for web and aggregated search, 2015.
12. Yuchen Zhang, Weizhu Chen, Dong Wang, and Qiang Yang. User-click modeling for understanding and predicting search-behavior. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, page 1388, 2011.

A Appendix

We give here some details about the inference in our TCM implementation outlined in section 2.8.

A.1 Click probability

For $P(F_{i,j} = 1)$ we introduce a variable $f_{i,j}$, which will be derived later.
By assumption that $M_i, E_{i,j}, R_{i,j}$and$F_{i,j}$ are independent, click probability can be formularize as:

$$\begin{aligned} P(C_{i,j} = 1) &= P(M_i = 1) * P(E_{i,j} = 1) * P(R_{i,j} = 1) * P(F_{i,j} = 1) && (21) \\ &= \alpha_1 * \beta_j * r_{i,j} * f_{i,j} && (22) \end{aligned}$$

A.2 Probability of the query match user intention

Because we remove equation that depends on α_2, we can now set α_1 as MLE.

$$P(M_i = 1) = \alpha_1$$

A.3 Probability of user submit next query

User submit next query if the query does not match user intention (α_1) or user want to search more.

$$\begin{aligned} P(N_i = 1) &= \frac{1}{|S|} \sum_{i \in S} \mathcal{I}(N_i = 1) \\ &= \frac{q_i}{|S|} \\ &= n_i \end{aligned}$$

q_i is the number of submitted-queries where user submit another query after i-th query session.

$$\begin{aligned} P(N_i = 1 | M_i = 1) &= \alpha_2 \\ &= \frac{P(N_i = 1) - P(N_i = 1 | M_i = 0) P(M_i = 0)}{P(M_i = 1)} \\ &= \frac{n_i + \alpha_1 - 1}{\alpha_1} \end{aligned}$$

A.4 Relevance probability

$$\begin{aligned} P(R_{i,j} = 1) &= r_{i,j} \\ &= \frac{\sum_{q_{i,j} \in S_{i,j}} P(R_{i,j} = 1 | C)}{|S_{i,j}|} \end{aligned}$$

Where $S_{i,j}$ are all sessions (queries) containing the document corresponding with the query i at rank j - document $P(R_{i,j} = 1|C)$ will be derive on eq.25

$$\begin{aligned} P(R_{i,j} = 1 | C) &= \mathcal{I}(C_{i,j} = 1) P(R_{i,j} | C_{i,j} = 1) + \mathcal{I}(C_{i,j} = 0) P(R_{i,j} | C_{i,j} = 0) && (23) \\ &= c_{i,j} + (1 - c_{i,j}) \frac{P(C_{i,j} = 0 | R_{i,j} = 1) P(R_{i,j} = 1)}{P(C_{i,j} = 0)} && (24) \\ &= c_{i,j} + (1 - c_{i,j}) \frac{P(C_{i,j} = 0 | R_{i,j} = 1) r_{i,j}}{1 - P(C_{i,j} = 1)} && (25) \end{aligned}$$

Where $c_{i,j} = 1$ if (i,j) was clicked in the current session. $P(C_{i,j} = 0|R_{i,j} = 1)$ is the chance of no click given that it is relevant.

$$
\begin{aligned}
P(C_{i,j} = 0|R_{i,j} = 1) &= P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1)P(M_i = 1) + P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0)P(M_i = 0) \\
&= \alpha_1 P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 1)P(E_{i,j} = 1) \\
&+ \alpha_1 P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 0)P(E_{i,j} = 0) \\
&+ (1 - \alpha_1)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 1)P(E_{i,j} = 1) \\
&+ (1 - \alpha_1)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 0)P(E_{i,j} = 0) \\
\\
&= \alpha_1\beta_j P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 1, F_{i,j} = 1)P(F_{i,j} = 1) \\
&+ \alpha_1\beta_j P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 1, F_{i,j} = 0)P(F_{i,j} = 0) \\
&+ \alpha_1(1 - \beta_j)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 0, F_{i,j} = 1)P(F_{i,j} = 1) \\
&+ \alpha_1(1 - \beta_j)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 0, F_{i,j} = 0)P(F_{i,j} = 0) \\
&+ (1 - \alpha_1)\beta_j P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 1, F_{i,j} = 1)P(F_{i,j} = 1) \\
&+ (1 - \alpha_1)\beta_j P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 1, F_{i,j} = 0)P(F_{i,j} = 0) \\
&+ (1 - \alpha_1)(1 - \beta_j)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 0, F_{i,j} = 1)P(F_{i,j} = 1) \\
&+ (1 - \alpha_1)(1 - \beta_j)P(C_{i,j} = 0|R_{i,j} = 1, M_i = 0, E_{i,j} = 0, F_{i,j} = 0)P(F_{i,j} = 0)
\end{aligned}
$$

We note that $P(C_{i,j} = 0|R_{i,j} = 1, M_i = 1, E_{i,j} = 1, F_{i,j} = 1) = 0$. Otherwise it is 1. From eq. 24 from TCM paper. Together with inserting our parameters this gives us the following:

$$
\begin{aligned}
P(C_{i,j} = 0|R_{i,j} = 1) &= \alpha_1\beta_j f_{i,j} + \alpha_1\beta_j(1 - f_{i,j}) + \alpha_1(1 - \beta_j)f_{i,j} + \alpha_1(1 - \beta_j)(1 - f_{i,j}) && (26) \\
&+ (1 - \alpha_1)\beta_j f_{i,j} + (1 - \alpha_1)\beta_j(1 - f_{i,j}) + (1 - \alpha_1)(1 - \beta_j)(f_{i,j} && (27) \\
&+ (1 - \alpha_1)(1 - \beta_j)(1 - f_{i,j}) && (28)
\end{aligned}
$$

expanding this we are only left with

$$
P(C_{i,j} = 0|R_{i,j} = 1) = 1 - (\alpha_1\beta_j f_{i,j}) \tag{29}
$$

Which seems intuitive as we assumed that all $M_i, R_{i,j}, E_{i,j}$ and $F_{i,j}$ are independent to get $P(C_{i,j} = 1)$. With this information we can calculate

$$
\begin{aligned}
P(R_{i,j} = 1|C) &= c_{i,j} + (1 - c_{i,j})\frac{(1 - (\alpha_1\beta_j f_{i,j}))r_{i,j}}{1 - \alpha_1\beta_j f_{i,j}r_{i,j}} && (30) \\
&= c_{i,j} + (1 - c_{i,j})\frac{r_{i,j} - \alpha_1\beta_j f_{i,j}r_{i,j}}{1 - \alpha_1\beta_j f_{i,j}r_{i,j}} && (31)
\end{aligned}
$$

A.5 Examination probability

$$
\begin{aligned}
P(E_{i,j} = 1) &= \beta_j && (32) \\
&= \frac{1}{|S|}\sum_{i\in S} P(E_{i,j} = 1|C) && (33)
\end{aligned}
$$

Where S is all sessions and i is a query within that session. $P(E_{i,j} = 1|C)$ will be derive on eq.34

$$
\begin{aligned}
P(E_{i,j} = 1|C) &= \mathcal{I}(C_{i,j} = 1)P(E_{i,j}|C_{i,j} = 1) + \mathcal{I}(C_{i,j} = 0)P(E_{i,j}|C_{i,j} = 0) && (34) \\
&= c_{i,j} + (1 - c_{i,j})\frac{P(C_{i,j} = 0|E_{i,j} = 1)P(E_{i,j} = 1)}{P(C_{i,j} = 0)} && (35) \\
&= c_{i,j} + (1 - c_{i,j})\frac{P(C_{i,j} = 0|E_{i,j} = 1)\beta_j}{1 - P(C_{i,j} = 1)} && (36)
\end{aligned}
$$

Where $c_{i,j}$ indicates whether document i,j was clicked. Analog to eq 29 we can show that

$$P(C_{i,j}=0|E_{i,j}=1)=1-(\alpha_1 f_{i,j} r_{i,j}) \tag{37}$$

This gives us

$$\begin{aligned} P(E_{i,j}=1|C) &= c_{i,j}+(1-c_{i,j})\frac{(1-(\alpha_1 f_{i,j} r_{i,j}))\beta_j}{1-\alpha_1\beta_j f_{i,j} r_{i,j}} && (38)\\ &= c_{i,j}+(1-c_{i,j})\frac{\beta_j-\alpha_1\beta_j f_{i,j} r_{i,j}}{1-\alpha_1\beta_j f_{i,j} r_{i,j}} && (39) \end{aligned}$$

A.6 Freshness probability

$$\begin{aligned} P(F_{i,j}=1|H_{i,j}=1) &= \alpha_3 && (40)\\ \alpha_3 &= \frac{1}{|S_{i,j}|}\sum_{q\in S}\sum_{(i,j)\in q} P(Fi,j=1|H_{i,j}=1,C) && (41) \end{aligned}$$

Where (i,j) is a query, rank pair identifying a certain document. $P(F_{i,j}=1|C)$ will be derive on eq.42 $P(F_{i,j}=1)$ will be derive on eq.48

$$\begin{aligned} P(F_{i,j}=1|H_{i,j}=1,C) &= \mathcal{I}(C_{i,j}=1)P(F_{i,j}=1|H_{i,j}=1,C_{i,j}=1) && (42)\\ &\quad+\mathcal{I}(C_{i,j}=0)P(F_{i,j}=1|H_{i,j}=1,C_{i,j}=0) && (43)\\ &= c_{i,j}+(1-c_{i,j})\frac{P(C_{i,j}=0|F_{i,j}=1,H_{i,j}=1)P(F_{i,j}=1|H_{i,j}=1)}{P(C_{i,j}=0|H_{i,j}=1)} && (44) \end{aligned}$$

Analog to eq 29 we can show that

$$P(C_{i,j}=0|F_{i,j}=1,H_{i,j}=1)=1-(\alpha_1\beta_j r_{i,j}) \tag{45}$$

We can also show

$$\begin{aligned} P(C_{i,j}=0|H_{i,j}=1) &= 1-P(C_{i,j}=1|H_{i,j}=1) && (46)\\ &= 1-(\alpha_1\alpha_3\beta_j r_{i,j}) && (47) \end{aligned}$$

The only difference between this and eq. 22 is that it is given that $H_{i,j}=1$ and because $H_{i,j}=1$ only has an influence on $P(F_{i,j}=1)$, namely that $P(F_{i,j}=1|H_{i,j}=1)=1$, we can substitute $f_{i,j}$ with $\alpha 3$ in eq. 22

Now we only need to calculate $f_{i,j}=P(F_{i,j})=1$

$$\begin{aligned} P(F_{i,j}=1) &= \mathcal{I}(H_{i,j}=1)P(F_{i,j}=1|H_{i,j}=1)+\mathcal{I}(H_{i,j}=0)P(F_{i,j}=1|H_{i,j}=0) && (48)\\ &= \mathcal{I}(H_{i,j}=1)\alpha_3+\mathcal{I}(H_{i,j}=0) && (49) \end{aligned}$$

Where $\mathcal{I}(H_{i,j}=1)$ is a binary indicator function from the data specifying whether document (i,j) was shown before in the current (q from eq. 41) session.

We could replace this indicator function with the probability that the document was examined the last time it was shown. This probability, called $H_{i,j}$ would depend on the probability that it was examined and $H_{i',j'}$ where i',j' is the last time this document was shown in the current session. It would look like this

$$P(H_{i,j}=1)=P(E_{i',j'}=1)P(H_{i',j'}=1) \tag{50}$$

then eq. 48 becomes:

$$
\begin{aligned}
P(F_{i,j}=1) &= P(H_{i,j}=1)\alpha_3 + P(H_{i,j}=0) && (51)\\
&= P(H_{i,j}=1)\alpha_3 + (1-P(H_{i,j}=1)) && (52)\\
&= \alpha_3 P(E_{i',j'}=1)P(H_{i',j'}=1) + (1-P(E_{i',j'}=1)P(H_{i',j'}=1)) && (53)
\end{aligned}
$$

Note that this discards the information that if (i',j') was clicked it surely was examined.

With eq 45 we can calculate $P(F_{i,j}=1|C)$

$$
\begin{aligned}
P(F_{i,j}=1|H=1,C) &= c_{i,j} + (1-c_{i,j})\frac{(1-(\alpha_1\beta_j r_{i,j}))\alpha_3}{1-\alpha_1\alpha_3\beta_j r_{i,j}} && (54)\\
&= c_{i,j} + (1-c_{i,j})\frac{\alpha_3-\alpha_1\alpha_3\beta_j r_{i,j}}{1-\alpha_1\alpha_3\beta_j r_{i,j}} && (55)
\end{aligned}
$$