Modeling multiple interactions with a Markov random field in query expansion for session search

Jingfei Li¹,³ | Xiaozhao Zhao¹ | Peng Zhang¹ | Dawei Song¹,²

¹Tianjin Key Laboratory of Cognitive Computing and Application, School of Computer Science and Technology, Tianjin University, Tianjin, China
²Department of Computing and Communications, The Open University, Milton Keynes, UK
³National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing, China

Abstract

How to automatically understand and answer users’ questions (e.g., queries issued to a search engine) expressed with natural language has become an important yet difficult problem across the research fields of information retrieval and artificial intelligence. In a typical interactive Web search scenario, namely, session search, to obtain relevant information, the user usually interacts with the search engine for several rounds in the forms of, e.g., query reformulations, clicks, and skips. These interactions are usually mixed and intertwined with each other in a complex way. For the ideal goal, an intelligent search engine can be seen as an artificial intelligence agent that is able to infer what information the user needs from these interactions. However, there still exists a big gap between the current state of the art and this goal. In this paper, in order to bridge the gap, we propose a Markov random field–based approach to capture dependence relations among interactions, queries, and clicked documents for automatic query expansion (as a way of inferring the information needs of the user). An extensive empirical evaluation is conducted on large-scale Web search data sets, and the results demonstrate the effectiveness of our proposed models.

KEYWORDS
Markov random field, multiple interactions, query expansion, session search
1 | INTRODUCTION

An ideal artificial intelligence (AI) system (eg, one that passes the Turing test) is expected to respond to users’ questions (or queries) naturally as a real human does. Achieving this ultimate goal involves obtaining as well as interacting and reasoning with information that is relevant to the questions, thus requiring a synergy across the research fields of information retrieval (IR), AI, and human-computer interaction (HCI). A large number of IR models are proposed to retrieve relevant information from large-scale web or local data repositories. An intelligent IR system can be seen as an AI agent1,2 that allows the user to continuously interact with the system and is able to automatically infer the user’s hidden information need from these interactions, eg, through query reformulations.

In an interactive Web search scenario, users usually interact with the search engine many times in order to accomplish a complex search task. This typical interaction process can be seen as a search session.1-3 Contrary to the traditional ad hoc search, session search allows IR models to retrieve documents by utilizing the historical interaction information within the same session. As illustrated in Figure 1, in a user’s search session, there exists a sequence of interactions in multiple forms (eg, query reformulations, clicks, and skips), which can be viewed as the user’s implicit relevance feedback. For example, the current query reflects the current information need directly (which may be insufficient and need to be refined), clicks may imply that users are interested in the viewed content4, the skip over some results may be a strong signal reflecting that the corresponding documents are irrelevant, reformulations of queries in a session imply that the user wants to search for some novel topics (reflecting an evolving information need) or enhance the queries’ representation power for the current information need, the whole session reflects the evolution of the user’s information need in response to the interactions, and so on. It is important to note that these interactions are usually mixed and intertwined with each other in a complex way (eg, clicking on a document may lead to skipping another document or a query modification), making the session search a difficult task. Thus, it is crucial to model and exploit such complex interactions and their interdependence relations in order to predict the user’s hidden search intent.

Query expansion, as an important means to represent users’ hidden search intent,5,6 expands the user’s original query by selecting relevant terms from a series of feedback documents (eg, through pseudo relevance feedback that simply assumes the top-ranked documents as relevant or estimated based on the user interactions as implicit relevance feedback as described above). A representative query expansion method is the relevance model (RM),7 which selects expansion words by considering the likelihood of relevance between original query and feedback documents. Later, a positional relevance model (PRM)8 selects expansion words that are focused on the
Session search allows information retrieval models to retrieve documents by utilizing an entire session, denoted as $S = \{I_1, I_2, \ldots, I_n, Q\}$, where $I_i$ is an interaction unit that contains a series of behaviors, e.g., skip (irrelevant results), click (relevant results), and query reformulations, and $Q$ is the current query of a session query topic, based on their positions and proximities to the query terms in feedback documents. The existing query expansion models do not explicitly consider 2 types of dependence relationships when assigning weights to expansion terms. These relationships are (i) between query terms and multiple interaction behaviors and (ii) between feedback documents and interaction behaviors.

In this paper, we propose to utilize the Markov random field (MRF) as a unified framework to model the dependence relations among different interaction events in a search session, based on which 3 query expansion models are derived. Different from the traditional query expansion models that only consider the relationship between feedback documents and original query, we introduce an additional variable to represent multiple interactions. Specifically, we explicitly consider 3 types of information in the process of query expansion. They are the current query $Q$, the feedback documents $D$ (including implicit relevance feedback documents and pseudo relevance feedback documents), and the interaction behaviors $I$ (including skips, clicks, and reformulations).

Before building an expanded query model, we first need to obtain a candidate set of words for expansion. For each candidate expansion word, we compute its weight based on the MRF model constructed from the interaction data. A series of feature functions is proposed for the MRF framework, so that different interaction information in search sessions is captured. We systematically investigate 3 different MRF models that are respectively underpinned by 3 different dependence assumptions (see Figure 2). The full independence model (FIM) considers that $Q, D,$ and $I$ determine the weight of an expansion word independently and assumes that the importance of all feedback documents $D$ is uniform (see Figure 2A). The query-document dependence model (also called the partial dependence model [PDM]) assumes that the selection of feedback documents should be dependent on the original query (see Figure 2B). Finally, in the full dependence model (FDM), we assume that the selection of feedback documents is also dependent on the dynamic interaction behaviors (see Figure 2C).

**FIGURE 2** Dependence graphs for 3 weighting functions based on different dependence assumptions between the components of the mixed feedback. $w$ is a word, $Q$ is the current query, $I$ is the interaction behaviors, and $D$ is a feedback document (a clicked document or a pseudo feedback document)
After obtaining the expansion terms for each session, we run the expanded Indri* query language in the Indri search engine to obtain the final search results. We have conducted an extensive empirical evaluation on the Session Track data in TREC (Text REtrieval Conference†) 2013 and 2014. The evaluation results demonstrate the effectiveness of the proposed models in comparison with a number of state-of-the-art baselines.

In a nutshell, the main contributions of this paper can be summarized as follows:

• We proposed a framework based on MRF that can model the dependence relationships between interaction behaviors, current query, and feedback documents in query expansion for session search.
• We proposed a series of feature functions for the MRF models, so that diversified interaction information can be captured within a unified framework.
• We conducted extensive comparative experiments on large-scale session search benchmarking data sets and demonstrated the effectiveness of our proposed models.

The rest of this paper is organized as follows. Section 2 reviews the related work. The proposed query expansion models are described in Section 3. Extensive evaluations are conducted in Section 4. In Section 5, we draw conclusions about this paper and discuss some possible directions for future work.

2 | RELATED WORK

Typical query expansion models can be based on explicit and implicit relevance feedback,9-12 pseudo relevance feedback,7,8,13-17 and external knowledge.18-22 In this paper, we focus on query expansion methods based on implicit and pseudo relevance feedback.

The implicit feedback methods can be categorized into 2 directions: query log based and HCI based. For example, Cui et al proposed query expansion models based on user interactions recorded in user logs.11,12 They selected high-quality expansion terms according to the correlations between query terms and document terms extracted by analyzing query logs. Chirita et al expanded the short Web search queries with terms collected from each user’s personal information repository, in order to resolve the ambiguity of the Web search queries and personalize the search results.10 Gao et al proposed a unified query expansion framework based on query logs using the path-constrained random walks.23 Joachims et al examined the reliability of implicit feedback generated from clickthrough data and query reformulations in WWW search and concluded that clicks are informative but biased. In addition to the log-based implicit feedback, there have been attempts to utilize HCI information to enhance the query expansion models. For example, Buscher et al used eye-tracking data to keep track of document parts that the user reads, and then the information at the subdocument level is used as implicit feedback for query expansion and document re-ranking.9 More recently, Chen et al have proposed a query expansion model based on the real-time reading content captured by an eye tracker.

Pseudo relevance feedback–based query expansion assumes that the top-ranked documents from a search engine are relevant. Rocchio proposed a classical query expansion model based on pseudo-relevant documents for the SMART retrieval system.15 After that, a series of pseudo relevance feedback–based models emerged. For example, Lavrenko and Croft proposed the

---

*https://sourceforge.net/projects/lemur/
†http://trec.nist.gov/*
well-known RM to estimate a language model from feedback documents, which can be used to estimate the weights of expanded terms. RM3, a further variant of RM, interpolates the term weights in an RM with that in the original query language model. The traditional pseudo feedback approaches utilize the whole feedback documents to extract words for query expansion, which may contain considerable irrelevant information. To solve this problem, retrieval models based on subdocument (e.g., passages) or term positions have been proposed. Similarly, Miao et al exploited the proximity between candidate expansion terms and query terms in the process of query expansion. Another direction toward improving the performance of pseudo feedback models is to select more reliable pseudo documents. For example, Lee et al presented a cluster-based resampling method to select better pseudo-relevant documents based on the RM. Miao et al integrated the topic space into pseudo relevance feedback in order to measure the reliability of the feedback documents. Ye and Huang evaluated the quality level of pseudo feedback documents with the Learning-to-Rank approach in pseudo relevance feedback.

In this paper, we select expanded terms from both clicked documents (implicit relevance feedback documents) and top-ranked documents in initial retrieval results (pseudo relevance feedback documents). The clicked documents in previous queries can reveal the information needed in a current search session, and the pseudo feedback documents can reflect the information needed for the current query to some extent. The weighting functions for expanded terms will take into account both interaction behaviors in the same session and the relevance of pseudo feedback documents. In our model, the position information is also considered, inspired by the positional language model (PLM) and the PRM. The dependence relationships among mixed and multiple types of interactions are modeled in the principled framework of the MRF, which has been successfully applied in IR. For example, Metzler and Croft modeled the term dependence relations in queries when ranking documents with MRF and then utilized it to model the term dependence relations in query expansion. In addition, a previous work on MRF models for session search has been proposed in order to capture the relationship between document and interactions. The MRF models in this paper are proposed in order to capture the relationship between expansion terms and multiple interaction behaviors.

### 3 | INCORPORATING MULTIPLE INTERACTIONS IN MRF FOR QUERY EXPANSION

In this section, we first present a framework, based on the MRF, to estimate a query expansion model for session search, namely, a session language model (SLM). Then, we formalize a series of feature functions based on 3 dependence assumptions. More details for the parameter estimation are also given.

#### 3.1 | An MRF-based SLM framework

We propose to estimate an SLM $\theta_S$ with the MRF, based on which we can generate the needed expanded terms $w$. According to Figure 1, we can estimate the SLM with the current query and the interaction behaviors. The estimation framework of the SLM is formalized as follows:

$$P(w|\theta_S) = P(w|Q, I) \propto P(w, Q, I) = \sum_{D \in \mathcal{P}} P(w, Q, I, D), \quad (1)$$
where $Q$ and $I$ are the current query and the interaction behaviors, respectively, and $D \in F$ is a feedback document (clicked document or pseudo feedback document). Now, the estimation of the SLM becomes a problem of estimating the joint probability $P = P(w, Q, I, D)$. Note that in this paper, we distinguish the interaction behaviors $I$ from the feedback documents $D$. $I$ is particularly focused on the general “behaviors” (eg, skip the irrelevant results, click on the possibly relevant documents, and query reformulations) rather than on a specific text document.

In order to estimate the joint probability $P(w, Q, I, D)$, we construct an MRF based on each of the dependence graphs $G$ in Figure 2. Each node in the graph represents a random variable. Particularly, the random variables are mutually independent if there is no edge between them. Therefore, we can make different dependence assumptions by deploying the corresponding edge configurations in the MRF graph, which will be presented in the following subsection in more detail. In this framework, the MRF graph $G$ contains 4 nodes, ie, $w$, $I$, $D$, and $Q$. The joint probability distribution over the 4 random variables is defined as follows, similarly as in the work of Metzler and Croft$^{33}$:

$$P_\Lambda = P_\Lambda (w, Q, I, D) = \frac{1}{Z_\Lambda} \prod_{c \in C(G)} \phi (c; \Lambda),$$

where $C(G)$ is the set of cliques in the MRF graph $G$, $\phi (c; \Lambda) \geq 0$ is a potential function over a clique $c$ ($c$ includes a series of nodes that are fully connected by edges between each other; see Figure 3), $\Lambda$ is a series of parameters that need to be estimated, and $Z_\Lambda = \sum_{w, Q, I, D} \prod_{c \in C(G)} \phi (c; \Lambda)$ is a normalization factor. However, it is generally infeasible to compute $Z_\Lambda$, since the number of terms in the summation is extremely large. To address this issue, we utilize an exponential function to guarantee the nonnegative property of the function (as in the work of Metzler and Croft$^{33}$), formalized as follows:

$$\phi (c; \Lambda) = \exp \left[ \lambda_c f_c (c) \right],$$

where $f_c (c)$ is a real-valued feature function over the clique $c$, and $\lambda_c$ is the importance weight for a specific feature function. Substituting this potential function back into Equation 2, we obtain the following joint probability distribution:

$$P_\Lambda \propto \exp \left[ \sum_{c \in C(G)} \lambda_c f_c (c) \right].$$

3.2 Variants of MRF

In this section, we describe and analyze 3 variants of the MRF model underpinned by 3 different dependence assumptions, respectively. The full independence (FI) assumption considers that $Q$, $I$, and $D$ are independent of each other and so is the generation of an expanded term $w$ from them.
respectively. The partial dependence (PD) assumption (namely, the query-document dependence assumption) assumes that the selection of a feedback document should depend on the current query. This is similar to the underlying idea for the classic RM (RM1),\textsuperscript{26} which assigns each feedback document a weight with the query likelihood. The full dependence (FD) assumption assumes that the selection of a feedback document is dependent on both the current query and the previous interaction behaviors. These 3 assumptions lead to 3 variants of MRF, which are detailed next.

### 3.2.1 Full independence model

Underpinned by the FI assumption, we can construct the MRF model FIM (see Figure 3A), in which the 3 nodes, corresponding to the current query \(Q\), interaction behaviors \(I\), and feedback document \(D\), are independent of each other when \(w\) is known. It contains 3 cliques, ie, \{\(I, w\)\}, \{\(D, w\)\}, and \{\(Q, w\)\}. Their corresponding feature functions are defined as follows:

\[
\begin{align*}
    f_Q(Q, w) &= \log P(w|Q)P(Q) \propto \log P(w|Q) \\
    f_D(D, w) &= \log P(w|D)P(D) \propto \log P(w|D) \\
    f_I(I, w) &= \log P(w|I)P(I) \propto \log P(w|I)
\end{align*}
\]

where \(P(Q)\), \(P(I)\), and \(P(D)\) are removed from the feature functions, since they can be regarded as certain events that have occurred (thus, \(P(Q) = 1\), \(P(I) = 1\), and \(P(D) = 1\)). In the proposed feature functions, \(f_Q(Q, w)\) quantifies how likely the word \(w\) and query terms co-occur in the same documents, \(f_D(D, w)\) measures the probability of \(w\) occurring in the feedback document \(D\), and \(f_I(I, w)\) models the possibility that the user utilizes \(w\) to represent the current information need in mind. Substituting these 3 feature functions back into Equation 4, we obtain the following joint probability distribution:

\[
P_\Lambda = \exp \left[\lambda_Q f_Q(Q, w) + \lambda_D f_D(D, w) + \lambda_I f_I(I, w)\right],
\]

where \(\lambda_Q\), \(\lambda_D\), and \(\lambda_I\) are 3 positive free parameters that satisfy condition \(\lambda_Q + \lambda_D + \lambda_I = 1\). In Section 3.3, we will describe the computation details for these feature functions.

### 3.2.2 Partial dependence model

The FI assumption in the previous model has an obvious limitation, since intuitively, the selection of feedback documents should depend on the original query. Therefore, we can add an edge between the current query node \(Q\) and the feedback document node \(D\). The motivation is to reward the expanded terms from the feedback documents that are more relevant to the original query. Compared with FIM, PDM contains one more clique \{\(Q, D, w\)\}. We define its feature function as follows:

\[
f_{Q,D}(Q, D, w) = \log P(w|Q, D)P(D|Q)P(Q) \propto \log P(w|Q, D)P(D|Q).
\]

In this feature function, \(P(D|Q)\) is the relevance probability of feedback document \(D\) with respect to the original query \(Q\). \(P(w|Q, D)\) is the probability that the document and the query jointly generate the expansion terms \(w\), which rewards the terms that are close to the query terms in the documents. Overall, the joint probability distribution estimated with this PDM can be formalized as follows:

\[
P_\Lambda = \exp \left[\lambda_Q f_Q(Q, w) + \lambda_D f_D(D, w) + \lambda_I f_I(I, w) + \lambda_Q f_{Q,D}(Q, D, w)\right],
\]
where the positive free parameters should satisfy condition $\lambda_Q + \lambda_D + \lambda_I + \lambda_{Q,D} = 1$. This model can integrate more dependence information than the FIM. The computation details will be described in Section 3.3.

### 3.2.3 Full dependence model

The FDM considers that the selection of feedback documents is dependent on both the original query and the previous interaction behaviors. Some feedback documents are relevant to the users’ information need for the current session, whereas some others are not. Intuitively, the relevant feedback documents should have an influence on the selection of expanded terms. However, the PDM method only uses the original query to estimate the relevant degree of a feedback document. To estimate the importance of each feedback document more precisely and generate more reliable expansion terms, we further improve the MRF model by adding another edge between $I$ and $D$. In this way, richer interaction information is integrated into the model. Accordingly, a new clique $\{I, D, w\}$ is brought into the MRF graph. For the added clique, we define its feature function as follows:

$$ f_{I,D}(I, D, w) = \log P(w|I, D)P(D|I)P(I) \propto \log P(w|I, D)P(D|I), $$

where $P(D|I)$ is the relevance probability of feedback documents estimated based on the interaction information, and $P(w|I, D)$ is the generative probability given the interactions and the feedback documents. In this way, we can obtain a new joint probability distribution based on the feature functions over all cliques in the FDM graph, as follows:

$$ P_A = \exp \left[ \lambda_Q f_Q(Q, w) + \lambda_D f_D(D, w) + \lambda_I f_I(I, w) + \lambda_{Q,D} f_{Q,D}(Q, D, w) + \lambda_{I,D} f_{I,D}(I, D, w) \right]. $$

Similarly, the positive free parameters should satisfy condition $\lambda_Q + \lambda_D + \lambda_I + \lambda_{Q,D} + \lambda_{I,D} = 1$. The computation of the feature functions will be detailed in Section 3.3.

### 3.3 Computation for feature functions

To compute all feature functions defined in the previous sections, we need to estimate the following probabilities: $P(w|Q)$, $P(w|D)$, $P(w|I)$, $P(D|Q)$, $P(w|Q, D)$, $P(D|I)$, and $P(w|D, I)$.

#### 3.3.1 Estimating $P(w|Q)$, $P(w|D)$, and $P(D|Q)$

The conditional probability of term $w$ conditioned on query $Q$ is computed as follows:

$$ P(w|Q) = \lambda \frac{co_{-}df(w, q_1, \ldots, q_n)}{co_{-}df(q_1, \ldots, q_n)} + (1 - \lambda) \sum_{i=1}^{n} \delta_i \frac{co_{-}df(w, q_i)}{df(q_i)}, $$

where $q_1, \ldots, q_n$ are query terms, $co_{-}df(t_1, \ldots, t_n)$ is the co-occurrence frequency of terms $t_1, \ldots, t_n$, and $df(t)$ is the document frequency of term $t$. Moreover, in the equation, the first item is the main part, the second item is the smoothing part, $\lambda$ is the smoothing parameter (in this paper, $\lambda = 0.8$), $\delta_i$ is the importance weight of a query term in the query, $\delta_i = \frac{tf_{idf}(q_i)}{\sum_{i=1}^{n} tf_{idf}(q_i)}$, $tf_{idf}(t) = tf(t) \times \log \frac{N_C}{df(t)}$, $tf(t)$ is the term frequency of term $t$ in the collection, and $N_C$ is the total number of documents in the collection.

The probability of $w$ occurring in a feedback document $D$ is estimated as follows:

$$ P(w|D) = \frac{tf(w, D) + \mu P(w|C)}{|D| + \mu}, $$

where $t_{\mu}$ is the total number of documents in the collection.
where \( tf(w, D) \) is the term frequency of \( w \) in \( D \), \( P(w|C) \) is the prior probability of \( w \) occurring in the collection, and \( \mu = 2500 \) is the smoothing parameter.

On the basis of Bayes's rule and the law of total probability, the relevance probability of \( D \) to the original query \( Q \) can be estimated as follows:

\[
P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)} = \frac{P(Q|D)P(D)}{\sum_{d \in \mathcal{D}} P(Q,d)} = \frac{P(Q|D)P(D)}{\sum_{d \in \mathcal{D}} P(Q|d)P(d)} \propto \frac{P(Q|D)}{\sum_{d \in \mathcal{D}} P(Q|d)},
\]

(13)

where each prior probability of feedback documents \( P(d) \) or \( P(D) \) is assumed to be uniform, \( P(Q|D) = \prod_{i=1}^{n} P(q_i|D) \propto \sum_{i=1}^{n} \log P(q_i|D), \) \( n \) is the number of words in the current query \( Q \). The final formula can be regarded as the normalized query likelihood of each feedback document based on all feedback documents.

### 3.3.2 Estimating \( P(w|Q, D) \)

Inspired by the PLM\(^3\) and the PRM,\(^8\) we develop a positional estimation method for the probability of a word \( w \) conditioned on the joint original query \( Q \) and feedback document \( D \). This model rewards the expanded terms that are within a closer proximity to the query terms in feedback documents. The estimation method is shown as follows:

\[
P(w|Q, D) = \frac{\sum_{i=1}^{|D|} c(w, i) \cdot \sum_{j=1}^{|Q|} \delta_j \cdot \exp \left[ \frac{-|i-p_j|^2}{2\sigma^2} \right] + \mu P(w|C)}{|D| + \mu},
\]

(14)

where \( i \) is an absolute position in the document \( D \), \( c(w, i) \in \{0, 1\} \) is the occurrence of term \( w \) in position \( i \); \( \delta_j \) is the importance weight of the \( j \)th query term in original query \( Q \), which is also defined in Equation 11; and \( p_j \) is the nearest position of query term \( q_j \) to the expanded term \( w \) in the document. Note that if \( q_j \) is not in the document, then \( p_j = i \). We follow the setting of \( \sigma = 200 \) as used in the work of Lv and Zhai.\(^8\) \( P(w|C) \) and \( \mu \) are the same as in Equation 12.

### 3.3.3 Estimating \( P(w|I) \)

The conditional probability \( P(w|I) \) models how likely a word \( w \) is generated conditioned on a user's rich interaction behaviors in the session. To compute the probability, we define 3 frequently observed behaviors in the users' interaction history, ie, skips, clicks, and query reformulations, formalized as \( I = \{\text{skip, click, QR}\}. \) QR is the abbreviation of query reformulations. Users usually skip some irrelevant results before deciding to click a document that seems relevant to the current information need after a series of query reformulations. Therefore, we regard “skip” as a kind of negative feedback that indicates the irrelevance of the skipped results, whereas we regard “click” as positive feedback indicating the relevance of clicked results. Different from the feedback information of “skip” and “click” at the document level, “query reformulations” can signal the relevance or irrelevance at the query terms level. On the basis of these intuitions and inspired by the query change model (QCM),\(^1,2\) we propose to model the conditional probability of \( w \) conditioned on the interaction behaviors as follows:

\[
P(w|I) = \sum_{i=1}^{n} \omega_i P(w|T_i),
\]

(15)

where \( T_i \) is a transition unit from the \( i \)th interaction unit to the \((i + 1)\)th interaction unit. Each \( T_i \) can stand for the transition between different interaction behaviors including “skip,” “clicks” in \( I_i \), and the “query reformulations” between \( Q_{i+1} \) and \( Q_i \), corresponding to \( I_{i+1} \) and \( I_i \). \( T_n \) is the
transition from the last interaction unit $I_n$ to the current query $Q$, and $n$ is the total number of the interaction units in the session. $P(w|T_i)$ is the conditional probability conditioned on the transition unit. $\omega_i = Z_n \log(1+i)$ is the discount factor for the $i$th transition unit, which penalizes the distant transition units to the current query, where $Z_n = 1/\sum_{i=1}^{n} \log(1+i)$ is the normalization factor. We further model the interaction behaviors in each transition unit to estimate the probability $P(w|T_i)$ as follows:

$$P(w|T_i) = \frac{1}{1 + \exp[P(w|D_{\text{skip}}) - P(w|D_{\text{click}})]} \times \left[ \alpha P(w|T_{i}^{\text{rmv}}) + \beta P(w|T_{i}^{\text{com}}) + \gamma P(w|T_{i}^{\text{add}}) \right],$$

(16)

where $T_{i}^{\text{rmv}}$, $T_{i}^{\text{com}}$, and $T_{i}^{\text{add}}$ respectively denote the removed query terms, the common query terms, and the added query terms, compared between $Q_{i+1}$ and $Q_i$, ie, the query reformulation information.

For example, suppose $Q_{i+1} = “abd”$ and $Q_i = “abc,”$ then the corresponding reformulations are $T_{i}^{\text{rmv}} = “c,”$ $T_{i}^{\text{com}} = “ab,”$ and $T_{i}^{\text{add}} = “d.”$ The conditional probability of $w$ conditioned on reformulation terms (ie, $T_{i}^{\text{rmv}}$, $T_{i}^{\text{com}}$, and $T_{i}^{\text{add}}$) can be estimated with Equation 11. $\alpha$, $\beta$, and $\gamma$ are the importance factors corresponding to 3 categories of reformulation terms. We should penalize the removed terms and reward the common terms and the added terms in different degrees when generating expanded terms, since query reformulations can reflect the trend of how a user’s search intent changes in the session.\(^1\,\,^{,2}\) To this end, we must guarantee that the conditions $\alpha < \beta < \gamma$ and $\alpha + \beta + \gamma = 1$ are satisfied.

In order to quantify the importance factors ($\alpha$, $\beta$, and $\gamma$), we first formalize their nonnormalized formulas, $\alpha’ = 1 - \sum_{t \in T_i^{\text{rmv}}} P(t|D_i)$, $\beta’ = 1 + \sum_{t \in T_i^{\text{com}}} P(t|D_i)$, and $\gamma’ = \max[\sum_{t \in T_i^{\text{add}}} \log \frac{N_c}{df(t)} \beta’]$, where $D_i$ is the concatenation (can be seen as a special document) of all snippets for the viewed search results in the $i$th interaction unit, and $P(t|D_i)$ is estimated with Equation 12. Then, they will be normalized, eg, $\alpha = \alpha’/\alpha’ + \beta’ + \gamma’$. The coefficient term before “$\times$” in Equation 16 models the positive and negative feedbacks indicated by “click” and “skip” behaviors. We will reward the expanded terms occurring in clicked snippets frequently and penalize those terms occurring in the skipped snippets frequently. $D_{\text{click}}$ and $D_{\text{skip}}$ denote the concatenations of all snippets for clicked results and skipped results. Note that if the user has not clicked any result, we will select the snippets for all nonclicked results to form $D_{\text{skip}}$. $P(w|D_{\text{click}})$ and $P(w|D_{\text{skip}})$ can be estimated with Equation 12.

### 3.3.4 Estimating $P(D|I)$

$P(D|I)$ measures the relevance probability of a feedback document $D$ given the interaction behaviors, under the assumption that a more relevant feedback document will have a greater influence on generating expanded query terms. For simplicity, we propose an approximation method for estimating the conditional probability, since we are concerned about the relative relevance between feedback documents. The formula is presented as follows:

$$P(D|I) = \frac{\text{Score}(D, I)}{\sum_{d \in F} \text{Score}(d, I)},$$

(17)

where $\text{Score}(\bullet, I)$ is the relevance score of a feedback document (represented with $\bullet$) given a series of interaction behaviors, and $F$ is the set of all feedback documents. Inspired by the idea of utilizing the whole session to score the retrieved documents in the work of Guan et al,\(^1\) we develop a
novel scoring function for modeling the complex interaction behaviors (i.e., skip, click, and query reformulations) in the whole session as follows (which is similar to Equations 15 and 16):

\[
\text{Score}(d, I) = \sum_{i=1}^{n} \omega_i \cdot \text{Score}(d, T_i) = \sum_{i=1}^{n} \omega_i \times \frac{1}{1 + \exp \left[ \text{sim}(d, D_{\text{skip}}) - \text{sim}(d, D_{\text{click}}) \right]}
\]

\[
\times \left[ \alpha \text{QL} (d, T_{i}^\text{run}) + \beta \text{QL} (d, T_{i}^\text{com}) + \gamma \text{QL} (d, T_{i}^\text{add}) \right]
\]

(18)

where the meanings and computation approaches for \(T_i\), \(\omega_i\), \(D_{\text{click}}\), \(D_{\text{skip}}\), \(T_{i}^\text{run}\), \(T_{i}^\text{com}\), \(T_{i}^\text{add}\), \(\alpha\), \(\beta\), and \(\gamma\) are the same as those in Equations 15 and 16. \(\text{sim}(d, \bullet)\) is the cosine similarity between a feedback document \(d\) and the special document \(D_{\text{click}}\) or \(D_{\text{skip}}\), in which all documents are represented with \(tf \times idf\) vectors. The function \(1/[1 + \exp(\bullet)]\) maps the reward values for “click” and the penalty values for “skip” into the interval of \((0, 1)\). We utilize the query likelihood function to compute \(QL(d, \bullet)\): specifically, \(QL(d, Q) = \prod_{t \in Q} P(t|d) \propto \sum_{t \in Q} \log P(t|d)\), where \(P(t|d)\) is estimated by the maximization likelihood approximation with Dirichlet smoothing, and the smoothing parameter is set as \(\mu = 1500\) empirically.

### 3.3.5 Estimating \(P(w|D, I)\)

Similar to \(P(w|I)\) and \(P(D|I)\), we also estimate \(P(w|D, I)\) by utilizing the whole session and considering the positive feedbacks, negative feedbacks, and all query reformulations, formalized as follows:

\[
P(w|D, I) = \sum_{i=1}^{n} \omega_i \cdot P(w|D, T_i) = \sum_{i=1}^{n} \omega_i \times \frac{1}{1 + \exp \left[ P(w|D_{\text{skip}}) - P(w|D_{\text{click}}) \right]}
\]

\[
\times \left[ \alpha P(w|T_{i}^\text{run}, D) + \beta P(w|T_{i}^\text{com}, D) + \gamma P(w|T_{i}^\text{add}, D) \right]
\]

(19)

where most parameters have appeared in previous equations (i.e., Equations 15 to 18). The conditional probability of \(w\) conditioned on query reformulations and a feedback document \(P(w|\bullet, D)\) can be estimated with Equation 14.

### 3.4 Strategies of parameter tuning

Given the formalized joint distribution and a set of feature functions, we should further tune the free parameters for each model, i.e., \(\lambda_Q, \lambda_D, \lambda_I, \lambda_{Q,D},\) and \(\lambda_{I,D}\). It is infeasible to obtain a globally optimized parameter configuration. To address this challenge, we develop an approximation algorithm to find the best parameter configurations in relatively small parameter spaces, which is shown in Figure 4 (Algorithm 1). This algorithm can reduce the parameter space greatly (the variable \(\text{step}\) controls the actual size of the parameter space), and the tuning speed will depend on the retrieval speed and the number of queries in the training set. Specifically, we first search the optimized parameter configuration \((\lambda_Q, \lambda_D, \lambda_I)\) for the FIM. Then, we control the relative ratio for the parameters of the FIM and search \(\lambda_{Q,D}\) for the PDM. Finally, we control the relative ratio for the parameters of the FDM and search for \(\lambda_{I,D}\).

### 4 Empirical Evaluation

We have developed 3 query expansion approaches for session search by modeling mixed interactions based on the MRF. To verify the effectiveness of the proposed models, we conduct extensive
LI ET AL.

FIGURE 4  Parameter tuning algorithm

experiments on the Session Track data of the TREC (Text REtrieval Conference) 2013 and 2014 with the ClueWeb12 Full corpus.

4.1  Experimental setup

The evaluation data sets are from TREC 2013 and TREC 2014. TREC released 87 session search tasks (sessions) in 2013 and 1021 tasks in 2014. However, given that the TREC 2014’s official ground truth only contains the first 100 sessions, we only select 100 sessions for TREC 2014 in our evaluations. In the ground truth, documents are labeled with graded relevance degrees (ie, $-2, 0, 1, 2, 3,$ and $4$, where $-2$ indicates the document is a spam document, $0$ stands for an irrelevant document, and $1-4$ represent the different relevance degrees of the document) with respect to the current query. Each search session includes a current query and a series of interaction units (see Figure 1), where each interaction unit records a historical query, the corresponding search results, and some interaction information (eg, skip, click, and dwell time). Session search allows to utilize the whole session to retrieve documents for the current query. We classify all search sessions into several classes according to the lengths of the current queries and the number of interaction units in sessions as shown in Table 1. From the Table, we can find that the current query lengths of most sessions fall in the interval between 2 and 5. Most sessions have 1 to 5 interaction units.

The document collection used in retrieval is ClueWeb12 Full corpus,$^\dagger$ which consists of 733 019 372 English web pages, collected between February 10, 2012 and May 10, 2012. We clean the ClueWeb12 corpus by filtering out the spam documents whose Waterloo Spam Ranking scores

\begin{algorithm}[H]
\caption{Parameter tuning algorithm.}
\begin{algorithmic}[1]
\State $C_{FIM} \leftarrow \emptyset$; // the parameter space for FIM
\State $C_{PDM} \leftarrow \emptyset$; // the parameter space for PDM
\State $C_{FDM} \leftarrow \emptyset$; // the parameter space for FDM
\For {$\lambda_Q = 0; \lambda_Q \leq 1; \lambda_Q \leftarrow \text{step}$}
\For {$\lambda_D = 0; \lambda_D \leq 1 - \lambda_Q; \lambda_D \leftarrow \text{step}$}
\State $\lambda_I \leftarrow 1 - \lambda_Q - \lambda_D$
\State Add configuration $c = \{\lambda_Q, \lambda_D, \lambda_I\}$ into $C_{FIM}$;
\EndFor
\EndFor
\For {$\lambda_Q, D = 0; \lambda_Q, D \leq 1; \lambda_Q, D \leftarrow \text{step}$}
\State $t \leftarrow 1 - \lambda_Q, D$; // $t$ is a temp value
\State $\lambda_Q \leftarrow \lambda_Q \times t$; $\lambda_D \leftarrow \lambda_D \times t$; $\lambda_I \leftarrow \lambda_I \times t$
\State Add $c = \{\lambda_Q, \lambda_D, \lambda_I, \lambda_Q, D\}$ into $C_{PDM}$;
\EndFor
\For {$\lambda_I, D = 0; \lambda_I, D \leq 1; \lambda_I, D \leftarrow \text{step}$}
\State $t \leftarrow 1 - \lambda_I, D$
\State $\lambda_Q \leftarrow \lambda_Q \times t$; $\lambda_D \leftarrow \lambda_D \times t$
\State $\lambda_I \leftarrow \lambda_I \times t$; $\lambda_Q, D \leftarrow \lambda_Q, D \times t$
\State Add $c = \{\lambda_Q, \lambda_D, \lambda_I, \lambda_Q, D, \lambda_I, D\}$ into $C_{PDM}$;
\EndFor
\For {$\lambda_Q, D, \lambda_I, D = 0; \lambda_Q, D, \lambda_I, D \leq 1; \lambda_Q, D, \lambda_I, D \leftarrow \text{step}$}
\State $t \leftarrow 1 - \lambda_Q, D, \lambda_I, D$
\State $\lambda_Q \leftarrow \lambda_Q \times t$; $\lambda_D \leftarrow \lambda_D \times t$
\State $\lambda_I \leftarrow \lambda_I \times t$; $\lambda_Q, D \leftarrow \lambda_Q, D \times t$
\State Add $c = \{\lambda_Q, \lambda_D, \lambda_I, \lambda_Q, D, \lambda_I, D\}$ into $C_{PDM}$;
\EndFor
\end{algorithmic}
\end{algorithm}

†http://www.lemurproject.org/clueweb12/index.php
TABLE 1  The distributions of the session number on current query length (CQLen) and the sessions’ interaction unit count (#I) for TREC (Text REtrieval Conference) 2013 and 2014

<table>
<thead>
<tr>
<th>CQLen</th>
<th>2013</th>
<th>2014</th>
<th>#I</th>
<th>2013</th>
<th>2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>20</td>
<td>19</td>
</tr>
<tr>
<td>2</td>
<td>17</td>
<td>17</td>
<td>2</td>
<td>18</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>34</td>
<td>3</td>
<td>14</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>22</td>
<td>4</td>
<td>6</td>
<td>26</td>
</tr>
<tr>
<td>5</td>
<td>16</td>
<td>9</td>
<td>5</td>
<td>7</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>8</td>
<td>6</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>7+</td>
<td>12</td>
<td>7</td>
<td>7+</td>
<td>16</td>
<td>9</td>
</tr>
<tr>
<td>#ALL</td>
<td>87</td>
<td>100</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

are less than 70. The corpus is indexed by Indri§ 5.6. In the indexing process, the stop words are removed, and all words are stemmed by porter Stemmer. Furthermore, we compare 7 retrieval models in our evaluations:

- **LM** (Baseline): the classical language model with Dirichlet smoothing; negative Kullback-Leibler divergence between the language models of query and document is used as the ranking function.
- **RM-PF**: the traditional relevance model based on pseudo feedback documents only; we reimplement it to expand the original query based on pseudo relevance feedback documents.
- **RM-MF**: a relevance model based on mixed feedback documents, including pseudo feedback documents and clicked documents in interaction history.
- **PRM-PF**: a positional relevance model based only on pseudo feedback documents; we reimplement it to expand the original query from pseudo relevance feedback documents.
- **PRM-MF**: a positional relevance model based on mixed feedback documents, including pseudo feedback documents and clicked documents in interaction history.
- **QCM**: a query change model for session search proposed in the work of Guan et al; we reimplement it as a re-ranking approach.
- **FIM**: full independence model.
- **PDM**: partial dependence model.
- **FDM**: full dependence model.

For all expansion models **RM-PF**, **RM-MF**, **PRM-PF**, **PRM-MF**, **FIM**, **PDM**, and **FDM**, the common free parameters are set to the same values. Specifically, when retrieving for a query, we apply the corresponding model to select 50 weighted terms from the feedback documents and expand the representation of the original query. The top 10 retrieved documents in the first round of search results with the language model are selected as pseudo feedback documents, since existing work has indicated that pseudo feedback models often gain the best performance when selecting about 10 pseudo relevance feedback documents. The second round of search results is obtained by running the expanded queries with the Indri search engine. TREC’s official evaluation metrics, i.e., normalized discounted cumulative gain (NDCG) and mean average precision (MAP), are adopted to evaluate the performance of the aforementioned retrieval models. Note that we compute the MAP based on the top N retrieved documents rather than all, namely, §https://sourceforge.net/projects/lemur/
In this paper, we report the MAP values based on different N values.

\[ \text{MAP} @ N = \frac{\sum_{q=1}^{Q} AP_q @ N}{Q}, \]

where \( Q \) is the number of tested queries and \( AP_q @ N = \frac{\sum_{k=1}^{N} (P_q(k) \chi_{rel(k)})}{N} \), where \( P_q(k) = |\{\text{relevant documents} \} \cap \{\text{top } k \text{ retrieved documents} \}|/k \), \( \chi_{rel(k)} \) is an indicator function equaling 1 if the item at rank \( k \) is a relevant document, zero otherwise. The difference between the definition of AP in this paper and the standard AP is that we use the number of the top N retrieved documents as the denominator rather than the number of all retrieved documents. In this paper, we report the MAP values based on different N values.

### 4.2 Evaluation results

In this section, we test different retrieval models with the corresponding optimal parameters (tuned in previous section) on large-scale data, ie, all Session Track Tasks in TREC 2013 and 2014 as reported in Table 1. The best parameter configurations for all tested models are summarized in Table 2. We tune the parameters for the proposed models (ie, FIM, PDM, and FDM) on different subsets of data. Specifically, we separate the TREC 2013 and 2014 data into 2 parts (ie, Part A and Part B) randomly. For TREC 2013, there are 44 sessions in Part A and 43 sessions in Part B. For TREC 2014, there are 50 sessions in 2 parts. We use one part as the training set (tuning parameters) and another as the testing set (using the trained parameter for testing). We apply Algorithm 1 described in Figure 4 to tune the parameters. In Table 2, we report the used parameters for each testing part. The overall average performance is analyzed respectively.

Table 3 reports the performance of retrieval models on TREC 2013 and 2014 evaluated with NDCG and MAP. The Table shows that all query expansion models and QCM outperform the baseline (LM), which demonstrates that the exploitation of session interaction information can significantly benefit the search performance.

#### Table 2: Optimal parameter configurations for different retrieval models

<table>
<thead>
<tr>
<th>Models</th>
<th>Optimal Parameter Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRM-PF</td>
<td>( \lambda = 0.1, \sigma = 200 ) (Lv and Zhai)</td>
</tr>
<tr>
<td>PRM-MF</td>
<td>( \lambda = 0.1, \sigma = 200 ) (Lv and Zhai)</td>
</tr>
<tr>
<td>QCM</td>
<td>( \alpha = 2.2, \beta = 1.8, \epsilon = 0.07, \delta = 0.4, \gamma = 0.92 ) (Guan et al)</td>
</tr>
</tbody>
</table>
| FIM    | TREC 2013 Part A: \( \lambda_Q = 0.2, \lambda_D = 0.16, \lambda_I = 0.64 \)
|        | TREC 2013 Part B: \( \lambda_Q = 0.4, \lambda_D = 0.36, \lambda_I = 0.24 \)
|        | TREC 2014 Part A: \( \lambda_Q = 0.2, \lambda_D = 0.64, \lambda_I = 0.16 \)
|        | TREC 2014 Part B: \( \lambda_Q = 0.2, \lambda_D = 0.16, \lambda_I = 0.64 \)
| PDM    | TREC 2013 Part A: \( \lambda_Q = 0.16, \lambda_D = 0.13, \lambda_I = 0.51, \lambda_{Q,D} = 0.2 \)
|        | TREC 2013 Part B: \( \lambda_Q = 0.32, \lambda_D = 0.29, \lambda_I = 0.19, \lambda_{Q,D} = 0.2 \)
|        | TREC 2014 Part A: \( \lambda_Q = 0.16, \lambda_D = 0.51, \lambda_I = 0.13, \lambda_{Q,D} = 0.2 \)
|        | TREC 2014 Part B: \( \lambda_Q = 0.04, \lambda_D = 0.03, \lambda_I = 0.13, \lambda_{Q,D} = 0.8 \)
| FDM    | TREC 2013 Part A: \( \lambda_Q = 0.26, \lambda_D = 0.23, \lambda_I = 0.15, \lambda_{Q,D} = 0.16, \lambda_{I,D} = 0.2 \)
|        | TREC 2013 Part B: \( \lambda_Q = 0.26, \lambda_D = 0.23, \lambda_I = 0.15, \lambda_{Q,D} = 0.16, \lambda_{I,D} = 0.2 \)
|        | TREC 2014 Part A: \( \lambda_Q = 0.26, \lambda_D = 0.23, \lambda_I = 0.15, \lambda_{Q,D} = 0.16, \lambda_{I,D} = 0.2 \)
|        | TREC 2014 Part B: \( \lambda_Q = 0.26, \lambda_D = 0.23, \lambda_I = 0.15, \lambda_{Q,D} = 0.16, \lambda_{I,D} = 0.2 \)

Abbreviations: FDM, full dependence model; FIM, full independence model; PDM, partial dependence model; PRM-PF, positional relevance model based on mixed feedback documents; PRM-MF, positional relevance model based only on pseudo feedback documents; QCM, query change model; TREC, Text REtrieval Conference.
TABLE 3  Overall performances for TREC 2013 and 2014 with respect to NDCG and MAP. Significance test has been done for different retrieval models compared with the RM3-PF model, where the symbol ‡ means $p < 0.01$ with the paired $t$ test and the symbol † means $p < 0.05$

<table>
<thead>
<tr>
<th>Models</th>
<th>NDCG@10</th>
<th>NDCG@100</th>
<th>MAP@10</th>
<th>MAP@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM</td>
<td>0.0570</td>
<td>0.0600</td>
<td>0.0071</td>
<td>0.0147</td>
</tr>
<tr>
<td>RM-PF</td>
<td>0.0951</td>
<td>0.0982</td>
<td>0.0178</td>
<td>0.0354</td>
</tr>
<tr>
<td>RM-MF</td>
<td>0.1042‡</td>
<td>0.113‡</td>
<td>0.0201†</td>
<td>0.0426‡</td>
</tr>
<tr>
<td>PRM-PF</td>
<td>0.1103‡</td>
<td>0.1137‡</td>
<td>0.0214†</td>
<td>0.046‡</td>
</tr>
<tr>
<td>PRM-MF</td>
<td>0.1114‡</td>
<td>0.1141‡</td>
<td>0.0211†</td>
<td>0.0457‡</td>
</tr>
<tr>
<td>QCM</td>
<td>0.1425‡</td>
<td>0.1299‡</td>
<td>0.0247†</td>
<td>0.0495‡</td>
</tr>
<tr>
<td>FIM</td>
<td>0.1418‡</td>
<td>0.1264‡</td>
<td>0.0268‡</td>
<td>0.0518‡</td>
</tr>
<tr>
<td>PDM</td>
<td>0.1396‡</td>
<td>0.1306‡</td>
<td>0.0265‡</td>
<td>0.051‡</td>
</tr>
<tr>
<td>FDM</td>
<td>0.1348‡</td>
<td>0.1302‡</td>
<td>0.0265‡</td>
<td>0.051‡</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Models</th>
<th>NDCG@10</th>
<th>NDCG@100</th>
<th>MAP@10</th>
<th>MAP@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM</td>
<td>0.1084</td>
<td>0.114</td>
<td>0.0202</td>
<td>0.0404</td>
</tr>
<tr>
<td>RM-PF</td>
<td>0.1247</td>
<td>0.1305</td>
<td>0.0230</td>
<td>0.0522</td>
</tr>
<tr>
<td>RM-MF</td>
<td>0.1246</td>
<td>0.1356‡</td>
<td>0.0231</td>
<td>0.0539†</td>
</tr>
<tr>
<td>PRM-PF</td>
<td>0.1407†</td>
<td>0.1517‡</td>
<td>0.0247</td>
<td>0.0629‡</td>
</tr>
<tr>
<td>PRM-MF</td>
<td>0.1392†</td>
<td>0.1541‡</td>
<td>0.0244</td>
<td>0.0637‡</td>
</tr>
<tr>
<td>QCM</td>
<td>0.1321</td>
<td>0.1450</td>
<td>0.0200</td>
<td>0.0553†</td>
</tr>
<tr>
<td>FIM</td>
<td>0.1658†</td>
<td>0.1590‡</td>
<td>0.0281†</td>
<td>0.0674‡</td>
</tr>
<tr>
<td>PDM</td>
<td>0.1661‡</td>
<td>0.1614‡</td>
<td>0.0286‡</td>
<td>0.0679‡</td>
</tr>
<tr>
<td>FDM</td>
<td>0.1638†</td>
<td>0.1605‡</td>
<td>0.0284‡</td>
<td>0.0676‡</td>
</tr>
</tbody>
</table>

Abbreviations: FDM, full dependence model; FIM, full independence model; LM, classical language model; MAP, mean average precision; NDCG, normalized discounted cumulative gain; PDM, partial dependence model; PRM-MF, positional relevance model based on mixed feedback documents; PRM-PF, positional relevance model based on pseudo feedback documents; QCM, query change model; RM-MF, relevance model based on mixed feedback documents; RM-PF, traditional relevance model based on pseudo feedback documents only; TREC, Text REtrieval Conference.

Moreover, our models outperform other query expansion models (ie, RM and PRM) with respect to all evaluation metrics. Specifically, our models outperform the RM and PRM models on both data sets. From the Table, we can also find that our models are competitive with the state-of-the-art session search model, ie, QCM, with respect to most evaluation metrics (except for NDCG@10). According to the reported results, we find that, for TREC 2013, our proposed models are similar to QCM in terms of different evaluation metrics. For TREC 2014, our proposed models can significantly outperform QCM. This phenomena may have resulted from the different features (eg, query length and interaction unit count) of TREC 2013 and TREC 2014. Specifically, from Table 1, we find that the proportion of long sessions (ie, $#I \geq 3$) in TREC 2014 (69%) is larger than that in TREC 2013 (56.3%), which shows that our proposed models can better handle the long sessions than QCM by effectively modeling the dependence among different interactions.

Comparing between RM-PF and RM-MF, we find that the retrieval performances of RM-MF are better than that of RM-PF on both TREC 2013 and 2014 with respect to all evaluation metrics. This demonstrates that utilizing clicked documents as implicit feedback when expanding the orig-
nal query can improve the quality of expansion terms significantly. This also shows that “click” is one of the positive feedback interaction behaviors. An unexpected phenomenon is that PRM-MF fails to outperform PRM-PF consistently. The possible reason my be that PRM assigns weights for expanded terms considering the distance between the current query and expanded terms in the feedback documents. In historical clicked documents, the occurrence frequency of current query terms is small, which leads to that the weights of expanded terms in clicked documents are very small.

From Table 3, we find that the dependence models (PDM and FDM) are often superior to the independence model (FIM), which shows that modeling dependence relations among mixed interactions is effective for improving the retrieval performance. However, the FDM fails to outperform the PDM. The possible reason is that the FDM rewards or penalizes some wrong documents when selecting feedback documents compared with the PDM.

5 | CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a unified framework based on MRFs, for modeling and incorporating complex dependence relations between mixed interaction feedbacks, eg, skip, click, and query reformulations in search sessions, to estimate an SLM and then use it to expand the current query in session search. Based on the MRF, we presented 3 dependence assumptions, and correspondingly, 3 MRF variants are derived. Rich interaction information is captured by computing the feature functions of MRF variants. Extensive experiments have been carried out on 2 large-scale standard data sets. The results demonstrate that our models outperform 2 strong baselines (RM and PRM) significantly on most sessions. Moreover, the dependence models outperform the independence model significantly.

The experimental results have validated the importance of modeling the mixed interactions and their complex dependence relations in IR. In the future, more dependence cases could be considered in the MRF. The proposed models may be further improved by reducing free parameters and exploiting automatic parameter tuning methods, eg, machine learning. Additionally, we consider that efficiency is another important performance issue for our model, especially when the method is applied to real Web search settings. Basically, before the proposed models can be applied to Web search settings, we should improve the efficiency of the computation of required probabilities. For example, we can perform some complex probability computation offline. This is left as a key direction of our future work.

ACKNOWLEDGMENTS

This work was supported in part by the Chinese National Program on Key Basic Research Project (973 Program) under grants 2014CB744604 and 2013CB329304; by the Chinese 863 Program under grant 2015AA015403; by the National Natural Science Foundation of China under grants U1636203, 61772363, 61272265, and 61402324; by the Tianjin Research Program of Application Foundation and Advanced Technology under grant 15JCQNJC41700; and by the European Union’s Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie grant agreement 721321.

ORCID

Jingfei Li http://orcid.org/0000-0001-5931-9866
REFERENCES


362


How to cite this article: Li J, Zhao X, Zhang P, Song D. Modeling multiple interactions with a Markov random field in query expansion for session search. Computational Intelligence. 2018;34:345–362. https://doi.org/10.1111/coin.12154