Improved Latent Concept Expansion Using Hierarchical Markov Random Fields

Hao Lang
Inst. of Computing Technology
Chinese Academy of Sciences
Beijing, P.R. China
langhao@ymail.com

Donald Metzler
Information Sciences Institute
U. of Southern California
Marina del Rey, CA
metzler@isi.edu

Bin Wang
Inst. of Computing Technology
Chinese Academy of Sciences
Beijing, P.R. China
wangbin@ict.ac.cn

Jin-Tao Li
Inst. of Computing Technology
Chinese Academy of Sciences
Beijing, P.R. China
jtli@ict.ac.cn

ABSTRACT

Most existing query expansion approaches for ad-hoc retrieval adopt overly simplistic textual representations that treat documents as bags of words and ignore inherent document structure. These simple representations often lead to incorrect independence assumptions in the proposed approaches and result in limited retrieval effectiveness. In this paper, we propose a novel query expansion technique that models the various types of dependencies that exist between original query terms and expansion terms within a robust, unified framework. The proposed model is called Hierarchical Markov random fields (HMRFs), based on Latent Concept Expansion (LCE). By exploiting implicit (or explicit) hierarchical structure within documents, HMRFs can incorporate hierarchical interactions which are important for modeling term dependencies in an efficient manner. Our rigorous experimental evaluation carried out using several TREC data sets shows that our proposed query expansion technique consistently and significantly outperforms the current state-of-the-art query expansion approaches, including relevance-based language models and LCE.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation, Theory

Keywords

query expansion, term dependence, Markov random fields

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.
Copyright 2010 ACM 978-1-4503-0099-5/10/10 ...$10.00.

1. INTRODUCTION

User queries serve as sparse, underspecified representations of information needs. Poor query representations can easily contribute to undesirable retrieval effectiveness. To alleviate this problem, researchers have developed various automatic query expansion techniques, including Rocchio's algorithm [24], Local Context Analysis [28], model-based feedback [30], relevance-based language models [13], and Latent Concept Expansion [20].

Most previously proposed query expansion approaches utilize overly simple document representations. For example, a common assumption is that documents are represented as bags of words. This representation ignores the important relationships and dependencies that exist between terms within a document. It is also commonly assumed that documents are flat (i.e., have no internal structure). However, many documents, such as HTML and XML documents are indeed structured. Even documents with no explicit structure, such as newswire articles, have some implicit structure, in the form of sentences, paragraphs, and so on. Thus, these simple representations often lead to incorrect independence assumptions in the proposed approaches, which ultimately degrades retrieval effectiveness.

There are two common independence assumptions made by most previous query expansion approaches. The first is the *independence assumption between query terms.* For query expansion, it is crucial to identify good feedback documents from which useful expansion terms can be extracted [14, 9]. Previous research has shown that retrieval models that consider dependencies between query terms can estimate document relevance more accurately [19], and the expansion models that build upon these models can achieve significantly better retrieval effectiveness [20].

The other assumption is the *independence assumption between query terms and expansion terms.* Even if a document is relevant, it might be only partially related [10]. Models that do not consider the dependencies between query terms and expansion terms assume that a query term is equally associated with all the terms within a document, at least from a theoretical point of view. This assumption is unlikely to hold, especially when expansion involves long documents, which are more susceptible to topic drift [17].

We are interested in addressing the following questions: Is there a way to model the two aforementioned types of dependencies within a robust, unified framework? Are there any computational complexity issues when considering both of these term dependencies? Can the two orthogonal term dependencies be combined to help each other or do they provide redundant information? To our best knowledge, these questions have never been seriously studied before.

In this paper, we introduce a novel graphical model called Hierarchical Markov random fields (HMRFs) for jointly modeling the dependencies between query terms and the dependencies between query terms and expansion terms. HMRFs build upon the Latent Concept Expansion (LCE) framework, which makes use of the Markov random field model for information retrieval [19, 20]. LCE is the first formalized query expansion approach that provides a mechanism for modeling term dependencies during expansion. This paper attempts to address one of LCE's limitations, that is, the assumption that query terms and expansion terms are conditionally independent given a document.

HMRFs represent documents using a simple tree structure instead of as flat and unstructured. As an undirected model, HMRFs relax the independence assumptions made by directed models (e.g., [13]) and can incorporate arbitrary features in a principled way. By exploiting implicit (or explicit) hierarchical structure within documents, HMRFs can incorporate hierarchical interactions which provide a mechanism for modeling term dependencies in an efficient manner. Furthermore, HMRFs can learn the importance of term dependencies in a way that directly optimizes an underlying retrieval metric and avoids the issue of metric divergence [21].

Specifically, the three primary contributions of this paper are:

1. A natural generalization of LCE, called LCE_GE, that explicitly models the dependencies between query terms and expansion terms. However, LCE_GE suffers from high computational complexity and data sparseness issues, which demonstrates that the problem of modeling these types of dependencies is non-trivial and requires additional effort.

2. A novel graphical model called Hierarchical Markov random fields (HMRFs). The model relaxes the independence assumptions made by existing query models [30, 13, 20] and provides a way for efficiently considering term dependencies by seamlessly integrating the hierarchical document structure.

3. A novel query expansion technique by embedding HMRFs with LCE (LCE_HMRF). Large-scale experimental results show that LCE_HMRF consistently and significantly outperforms the current state-of-the-art query expansion approaches, which indicates the importance of considering term dependencies for query expansion.

The rest of this paper is organized as follows. In Section 2, we briefly review related work on query expansion. In Section 3, we present our extension of LCE that models dependencies between original query terms and expansion terms. In Section 4, we describe Hierarchical Markov random fields in details. Then, in Section 5, we show the time complexity of our proposed approaches in this work. In Section 6, we evaluate our proposed models and analyze the results. Finally, Section 7 concludes the paper and discusses some possible directions for future research.

2. RELATED WORK

Among all the techniques for improving query representations, query expansion is arguably among the most effective and widely used. One of the first approaches, that is still often used even today, is Rocchio's algorithm, proposed in 1971 for the SMART retrieval system [24]. Rocchio's algorithm was developed within the vector space retrieval model. It reweights the original query vector by moving the weights towards the set of relevant (or pseudo-relevant) documents and away from the non-relevant documents.

There has been some work on developing formalized query expansion techniques under the language modeling framework, including model-based feedback [30] and the relevance-based language models [13]. Both approaches use relevant (or pseudo-relevant) documents to estimate an improved query model. These models tend to ignore important issues such as term dependence, proximity, and document structure. The use of positional language models for query expansion may be useful for exploring some of these factors, but has yet to be formally examined [16].

Relevance-based language models, or simply *relevance models*, form the basis for LCE and our proposed extensions. Thus, we will now describe some of the details of the model. Given a query Q, a relevance model is a multinomial distribution over terms, $P(.|Q)$. The model is estimated as follows:

$$\begin{aligned} P(w|Q) &= \int_D P(w|D)P(D|Q) \\ &\approx \frac{\sum_{D\in\mathcal{R}_Q} P(w|D)P(Q|D)P(D)}{\sum_w \sum_{D\in\mathcal{R}_Q} P(w|D)P(Q|D)P(D)} \end{aligned} \qquad (1)$$

where $\mathcal{R}_Q$ is the set of documents that are relevant (or pseudo-relevant) to the query. In the setting of pseudo-relevance feedback, these are the top ranked documents for the query. Meanwhile, it is assumed that $P(D)$ is uniformly distributed.

After the model is estimated, the k terms with the highest likelihood from $P(.|Q)$ are extracted as expansion terms. These terms are then added to the original query and weighted according to their likelihood of being generated by the relevance model. We refer to this particular instantiation of the relevance model as RM3 [1].

The two approaches that are most related to our work account for term dependencies and document structure during query expansion. The first approach is Xu and Croft's Local Context Analysis (LCA) [28]. LCA combines passage-level retrieval with concept expansion, where concepts are single terms and phrases. However, the weights of concepts are estimated in a heuristic manner and it is unclear how much the phrases helped over the single terms alone. The second approach, that forms the basis of our work, is Metzler and Croft's Latent Concept Expansion [20], which is based on the Markov random field model [19]. The LCE approach provides a mechanism to model term dependencies and achieves superior retrieval effectiveness compared to state-of-the-art query expansion approaches, including relevance models. However, it has several limitations, as we described earlier.

Recently, Cao et al. attempted to improve query expansion effectiveness at the term level, by using term proximity features to learn a classifier for selecting useful expansion terms [5]. Meanwhile, He et al. attempted to improve the expansion effectiveness at the document level by detecting good feedback documents. They classified all feedback documents using a variety of features such as the distribution of query terms in the feedback document and proximity between the expansion terms and the original query terms in the feedback document [9].

In contrast to previous approaches, our proposed query expansion approach robustly models term dependencies, document structure, and arbitrary features all within a formally motivated framework. As we will show, this yields a highly useful query expansion model that achieves state-of-the-art retrieval effectiveness.

3. GENERALIZED LCE

In this section, we introduce a novel query expansion paradigm that generalizes LCE [20] for fully considering dependencies among original query terms and expansion terms. Before proposing our generalized version of LCE, we first provide an overview of the original LCE approach.

3.1 Latent Concept Expansion

Latent concept expansion (LCE) assumes that when a user formulates her original query, she has some set of concepts in mind, but is only able to express a small number of them in the form of a query. The concepts that the user has in mind, but did not express in the query, are called latent concepts. LCE attempts to recover these latent concepts, based on their co-occurrences with these concepts explicitly expressed in the original query, in the relevant or pseudo-relevant documents.

Specifically, LCE first constructs a Markov random field model consisting of the original query terms, the expansion concept, and the document node [20]. Figure 1 (middle) shows an example LCE MRF model that is used to expand a three word query ($Q = q_1\ q_2\ q_3$) with single term concepts (e_1). Throughout this paper we assume that all MRF models are based on the *sequential dependence* assumption, which assumes that dependencies exist between adjacent query terms [19]. We also only consider single term latent concepts, as multi-term concepts were not found to be useful for expansion in the original LCE study [20].

Given the graph, a joint distribution over a query (Q), a latent concept (E), and a document (D) can be defined. The conditional probability of a latent concept given the query can be easily computed as:

$$P(E|Q) \approx \frac{\sum_{D \in \mathcal{R}_Q} P(Q, E, D)}{\sum_E \sum_{D \in \mathcal{R}_Q} P(Q, E, D)} \tag{2}$$

where $P(Q, E, D)$ is the joint probability and $\mathcal{R}_Q$ is the set of relevant or pseudo-relevant documents for query Q. After computing the conditional probabilities, the k latent concepts E with the highest conditional probability are chosen as expansion concepts and an augmented MRF is constructed and used for retrieval.

3.2 Generalization

As illustrated in Figure 1 (middle), LCE assumes that query terms and latent concepts are conditionally independent given a document. We hypothesize that this independence assumption can degrade retrieval effectiveness, because it is important to expand the query with concepts that are actually related to the original query terms. Therefore, we extend the model structure of LCE, by adding links between each query term node and the latent concept node. We refer to this generalized model as LCE_GE. The graphical model representation of the model is shown in Figure 1 (right). With the newly added links, LCE_GE can explicitly model the dependencies between query terms and latent concepts.

Name	Description
T_D	set of cliques containing the document node and exactly one query term node or the latent concept node
O_D	set of cliques containing the document node and two or more query term nodes that appear in sequential order within the query
U_D	set of cliques containing the document node and two or more query term nodes that appear in any order within the query
T_{QE}	set of cliques containing the document node, the latent concept node, and exactly one query term node
O_{QE}	set of cliques containing the document node, the latent concept node, and two or more query term nodes that appear in sequential order within the query
U_{QE}	set of cliques containing the document node, the latent concept node, and two or more query term nodes that appear in any order within the query
D	set of cliques containing only the document node

Table 1: Summary of LCE_GE clique sets.

Following the original LCE approach, we parameterize the MRF model based on clique sets to provide more flexibility in encoding useful features over cliques in the graph while keeping the number of features and parameters reasonable [20]. Cliques in the same clique set can share feature functions and parameters, which significantly reduce the parameter space. Meanwhile, we can tune the parameters on the level of clique sets, which is more effective and efficient than tuning on the level of cliques for information retrieval. We expand upon the clique sets used by LCE and ultimately make use of seven clique sets, which are summarized in Table 1.

After tying the parameters based on the proposed clique sets, the joint distribution is defined as:

$$\begin{aligned} P(Q, E, D) = &\frac{1}{Z} \exp\{\lambda_{T_D} \sum_{c \in T_D} f_{T_D}(c) + \lambda_{O_D} \sum_{c \in O_D} f_{O_D}(c) + \\ &\lambda_{U_D} \sum_{c \in U_D} f_{U_D}(c) + \lambda_{T_{QE}} \sum_{c \in T_{QE}} f_{T_{QE}}(c) + \\ &\lambda_{O_{QE}} \sum_{c \in O_{QE}} f_{O_{QE}}(c) + \lambda_{U_{QE}} \sum_{c \in U_{QE}} f_{U_{QE}}(c) + \\ &\lambda_D f_D(D)\} \end{aligned} \tag{3}$$

To utilize the model, we must define the feature functions $f(c)$. One appealing property of the MRF model is its ability to use a variety of features for ranking. The correct choice of features depends largely on the retrieval task and the evaluation metric. It is unlikely there is a single, universally applicable set of features. Since it is not our goal here to

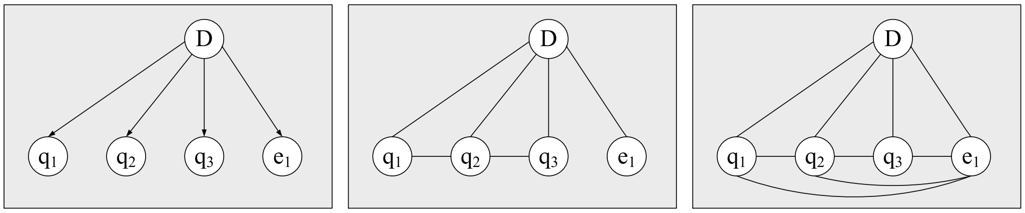

Figure 1: Graphical model representations of the relevance model (left), latent concept expansion (middle), and LCE_GE (right) for a three term query.

Feature	Value
$f_{T_D}(q, D)$	$\log[(1-\alpha)\frac{tf_{q,D}}{\lvert D\rvert} + \alpha\frac{cf_q}{\lvert C\rvert}]$
$f_{O_D}(b, D)$	$\log[(1-\beta)\frac{tf_{\#1(b),D}}{\lvert D\rvert} + \beta\frac{cf_{\#1(b)}}{\lvert C\rvert}]$
$f_{U_D}(b, D)$	$\log[(1-\beta)\frac{tf_{\#uw(b),D}}{\lvert D\rvert} + \beta\frac{cf_{\#uw(b)}}{\lvert C\rvert}]$
$f_{T_{QE}}(q, E, D)$	$\log[(1-\beta)\frac{tf_{\#uw(q,E),D}}{\lvert D\rvert} + \beta\frac{cf_{\#uw(q,E)}}{\lvert C\rvert}]$
$f_{O_{QE}}(b, E, D)$	$\log[(1-\beta)\frac{tf_{\#uw(b,E),D}}{\lvert D\rvert} + \beta\frac{cf_{\#uw(b,E)}}{\lvert C\rvert}]$
$f_{U_{QE}}(b, E, D)$	$\log[(1-\beta)\frac{tf_{\#uw(b,E),D}}{\lvert D\rvert} + \beta\frac{cf_{\#uw(b,E)}}{\lvert C\rvert}]$
f_D	0

Table 2: Feature functions used in MRF model. Here, q is a query term, b is a query bigram, $tf_{w,D}$ is the number of times term w occurs in document D, $tf_{\#1(b),D}$ denotes the number of times the exact phrase b occurs in document D, $tf_{\#uw(b),D}$ is the number of times the terms in b appear ordered or unordered within a window of 8 terms, and $|D|$ is the length of document D. The cf and $|C|$ values are analogously defined on the collection level. Finally, α and β are model hyperparameters that control smoothing for single term and phrase features, respectively.

find optimal features, we use a simple, fixed set of features that have shown to be effective in previous work [19, 20]. Table 2 lists the features used in Equation 3. These features attempt to capture various types of term occurrence and term proximity information.

It is important to note that relevance-based language models, LCE, and our generalized version of LCE are all closely connected. For example, notice the striking similarity in Equations 1, 2, and 3. The core difference between the approaches, as is clearly illustrated by Figure 1, is how the joint distribution $P(Q, D, E)$ is defined. Relevance models make the most assumptions (directed model, conditional independence between q_i and e), while generalized LCE makes the fewest (undirected model, no conditional independence between q_i and e).

Although our proposed LCE_GE approach can incorporate all of the dependencies essential for the query expansion task, its model structure may not be the most appropriate for the following reasons.

High computational complexity: In LCE_GE, the latent concept now explicitly depends on the user query. Since a latent concept could be any term in the document collection with some probability, there are large amount of pairs of query terms and latent concepts that must be evaluated. Moreover, evaluating term proximity features can be time consuming for most retrieval systems compared to evaluating term occurrence features, especially when the retrieval system uses standard positional inverted indexes. As a result, it may be practically infeasible to implement the LCE_GE approach in a large-scale retrieval system. We will provide a more formal analysis of the time complexity of LCE_GE in Section 5.

Data sparseness: Data sparseness refers to the fact that most query terms only have a few number of occurrences distributed in a relatively long document and hence a very sparse query term-latent concept relationship matrix is generated. This problem is also compounded by the fact that term proximity feature functions are often more sparse than term occurrence feature functions, since term co-occurrences must be within a window size (e.g., 8 terms in LCE_GE).

In the next section, we will introduce a different, implicit way of modeling dependencies between query terms and expansion concepts that can be computed more efficiently than LCE_GE while maintaining strong retrieval effectiveness.

4. LCE USING HIERARCHICAL MRFS

In this section, we describe our proposed Hierarchical Markov random field model, which will be used in conjunction with LCE to construct a novel, highly effective query expansion approach.

4.1 Data Representation

For the query expansion task, it is important to use a representation that makes full use of the document as evidence. Most previous approaches use unstructured bag of words document representations, and therefore can only explore co-occurrence features at the document level. This type of representation is too coarse for our needs, especially for long documents which are more susceptible to topic drift [17].

Instead, we need a representation that allows us to explore co-occurrence at different spatial scales within the document, such as sentence-, paragraph-, or passage-level scales. Therefore, we represent documents using a tree structure. Each node in the tree represents a content region within the document, with the root node representing the whole document. Each node is the aggregation of all its children nodes. All leaf nodes are basic content units and form a flat segmentation of the document.

We are interested in partitioning documents into topical segments. Some learning methods have recently been proposed to automatically infer document structure from unstructured documents [3, 11]. However, such methods may be difficult to implement, computationally expensive, or simply inappropriate for our given task. Therefore, to approx-

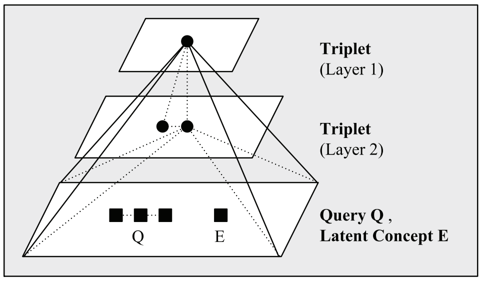

Figure 2: Illustration of LCE_HMRF structure for a three term query.

imate the effect of representing a document as a tree structure, we represent a document as a *two-layer tree*. The first layer contains a single root node representing the entire document, and the second layer contains nodes that partition the document into fixed length segments (passages). Thus, segments within the document are the basic content units now, instead of the entire document. In principle though, there is nothing preventing the model described in this section from extending beyond two layers or segmenting documents in some other way.

In this work, we are interested in triplets of nodes $\{S_{j-1}, S_j, D\}$ from the two-layer tree. Nodes S_{j-1} and S_j are associated with neighboring sibling nodes in the second layer of the document tree (i.e., they are adjacent segments), while node D is the root node of the tree (i.e., the entire document). Here, we use the positional information to sequentialize the nodes on the tree from left to right. The triplets extracted from a single document form a set and cover all the nodes of the document tree, but are not a partition, since some nodes appear in multiple triplets. The remainder of this section describes how MRFs can be used to model a joint distribution over a query, a latent concept, and this triplet, and how the distribution can ultimately be used for query expansion.

4.2 Model Description

Given the hierarchical data representation and the triplets extracted from the document tree, a Hierarchical Markov random field (HMRF) model can easily be constructed and used to generate latent concepts. When HMRFs are used in this way (i.e., within the LCE framework to generate latent concepts), we refer to the resulting approach as LCE_HMRF. Figure 2 provides a high level illustration of the LCE_HMRF model structure. We assume there are dependencies between nodes that are associated with parent and child nodes in the document tree. Meanwhile, we assume a dependency exists between nodes that are associated with neighboring sibling nodes at the same layer of the document tree. Query terms and the latent concept are under the same independence assumptions adopted by LCE, that is, neighboring query terms are linked, and both query terms and the latent concept are linked to all the nodes of the triplet. Note that there are *no* links between query terms and the latent concept as in LCE_GE.

Name	Description
T_{SQ}	set of cliques containing exactly one query term node, the root node at the first layer of the document tree, and two neighboring nodes at the second layer of the document tree.
T_{SE}	set of cliques containing the latent concept node, the root node at the first layer of the document tree, and two neighboring nodes at the second layer of the document tree.

Table 3: Two additional clique sets used with LCE_HMRF.

Feature	Value
$f_{T_{SQ}}(q, D^*)$	$\log[(1-\delta-\zeta)\frac{tf_{q,S_j}}{\lvert S_j\rvert}+\delta\frac{tf_{q,S_{j-1}}}{\lvert S_{j-1}\rvert}+\zeta\frac{tf_{q,D}}{\lvert D\rvert}]$
$f_{T_{SE}}(e, D^*)$	$\log[(1-\tilde{\delta}-\tilde{\zeta})\frac{tf_{e,S_j}}{\lvert S_j\rvert}+\tilde{\delta}\frac{tf_{e,S_{j-1}}}{\lvert S_{j-1}\rvert}+\tilde{\zeta}\frac{tf_{e,D}}{\lvert D\rvert}]$

Table 4: Document structure feature functions used with LCE_HMRF. Here, $D^* = (S_{j-1}, S_j, D)$, and δ, ζ, $\tilde{\delta}$, and $\tilde{\zeta}$ are model hyperparameters that control smoothing for query term and latent concept features, respectively.

We parameterize LCE_HMRF based on clique sets in the same way as LCE_GE. Unlike LCE_GE, LCE_HMRF can exploit different levels of contextual information in a document. We propose two novel clique sets, which provide a mechanism to exploit a document's fine-grained evidence. Features over these cliques encode how well the terms in the clique describe a basic content unit (i.e. a segment) within the document. The two new clique sets are described in Table 3. Meanwhile, LCE_HMRF can also exploit a document's coarse-grained evidence, like LCE_GE. That is, the clique sets T_D, O_D, U_D, D that are proposed for LCE_GE are also used here, where the original document node on the LCE_GE graph is replaced by the root node at the first layer of the document tree. Putting everything together, the joint distribution of LCE_HMRF can be written as:

$$P(Q, E, S_{j-1}, S_j, D) = \frac{1}{Z}\exp\{$$

$$\underbrace{\lambda_{T_D}\sum_{c\in T_D} f_{T_D}(c) + \lambda_{O_D}\sum_{c\in O_D} f_{O_D}(c) + \lambda_{U_D}\sum_{c\in U_D} f_{U_D}(c) +}_{F_D(Q)+F_D(E)\ -\ \text{document level score}}$$

$$\underbrace{\lambda_{T_{SQ}}\sum_{c\in T_{SQ}} f_{T_{SQ}}(c) + \lambda_{T_{SE}}\sum_{c\in T_{SE}} f_{T_{SE}}(c)}_{F_S(Q)+F_S(E)\ -\ \text{segment level score}} + \lambda_D f_D(D)\} \qquad (4)$$

where F_D and F_S are convenience functions defined by coarse-grained evidence oriented (at the document level) and fine-grained evidence oriented (at the segment level) components of the joint distribution, respectively. Furthermore, $F_D(Q)$ and $F_S(Q)$ are document and query dependent, while $F_D(E)$ and $F_S(E)$ are document and latent concept dependent. These will be used to simplify and clarify expressions derived throughout the remainder of this paper.

Table 4 shows how $f_{T_{SQ}}$ and $f_{T_{SE}}$ are defined. The feature function $f_{T_{SQ}}$ estimates the relevance of a basic content unit (i.e., a segment) within the document to the query. It is based on three hypotheses: 1) if a document is relevant to the query, then a segment within the document is more likely to be relevant, 2) if a segment's neighboring segments are relevant to the query, then the segment itself is more likely to be relevant, and 3) the more query term occurrences a segment has, more likely the segment is likely to be relevant to the query. This is similar to the smoothing used by Murdock and Croft for sentence retrieval, which was shown to be highly effective [22]. Additionally, the feature function $f_{T_{SE}}$ measures how relevant the latent concept is with respect to the segment in a similar manner.

4.3 Parameter Estimation

In Section 4.1 and Section 4.2, we introduced the data representation and model description of our proposed Hierarchical Markov random field model. In this section, we detail our method for estimating the model parameters.

The HMRF model is an extension of the Markov random field model. Although the HMRF model is an generative model, it is inappropriate to train the model using traditional likelihood-based approaches, such as maximum likelihood estimation and maximum a *posteriori* estimation. This is because our goal in leveraging the term dependency aware model is primarily for information retrieval, and thus retrieval metrics of interest are the key metrics for us to optimize for. Likelihood-based approaches are unlikely to maximize the retrieval metrics (e.g., mean average precision), due to the issue of *metric divergence* [21].

For this reason, we discriminatively train our model to directly optimize the parameters for the evaluation metric under consideration [26, 19, 2]. It is easy to see that the joint distribution in Equation 4 is *linear* with respect to the model parameters. Hence, we make use of the coordinate-level ascent algorithm that was original proposed in [18], which is easy to implement for a small number of parameters (as is the case here), and has good empirically verified generalization properties.

The coordinate-level ascent algorithm iteratively optimizes a multivariate object function by performing a series of one-dimensional line searches. It repeatedly cycles through each parameter in Equation 4, while holding all the other parameters fixed. This process is performed iteratively over all parameters until the gain for a given task is below a certain threshold.

We note that there is a large and growing body of literature on the learning to rank methods for information retrieval, which have been developed for effectively optimizing ranking functions with respect to ranking metrics [15]. Other more sophisticated learning to rank methods for linear models can also be used in this work, such as ranking SVM[12] and SVM^{MAP}[29]. Since our work focus on studying the importance of term dependencies for query expansion and our model parameter space is small, we employed a simple, yet effective parameter estimation method instead.

4.4 Discussion

Both LCE_GE and LCE_HMRF attempt to relax the independence assumption between query terms and latent concepts. LCE_GE *explicitly* incorporates the dependencies by linking query term nodes and the latent concept node on the graph. LCE_HMRF instead *implicitly* models these same dependencies via the segment nodes. In this way, the model can identify, and reward, expansion concepts that co-occur with one or more query terms within one or more segments. Modeling such co-occurrences on a per-segment basis, rather than at the document-level provides an implicit means for modeling dependencies between query terms and expansion concepts.

By implicitly modeling the dependencies, the latent concept in LCE_HMRF now depends on the hierarchical structure within documents rather than the user query as in LCE_GE. As a result, LCE_HMRF does not have to evaluate expensive features defined over pairs of query terms and latent concepts like LCE_GE. In this way, LCE_HMRF can address LCE_GE's high computational complexity and data sparseness problems.

To illustrate how LCE_HMRF weights expansion concepts, we show the general form of the LCE conditional probabilities, which take on the following form:

$$P(E|Q) \approx \frac{\sum_{S_{j-1},S_j,D} P(Q,E,S_{j-1},S_j,D)}{\sum_E \sum_{S_{j-1},S_j,D} P(Q,E,S_{j-1},S_j,D)} \propto \sum_{S_{j-1},S_j,D} exp\{F_D(Q)+F_D(E)+F_S(Q)+F_S(E)\} \tag{5}$$

where $\{S_{j-1}, S_j, D\}$ is the set of triplets extracted from the set of relevant or pseudo-relevant documents R_Q for query Q. As we see, the contribution for each triplet is a combination of scores at the document level and scores at the segment level. The document level score reflects how relevant the original query is to the document, as well as how relevant the expansion concept is. The same is true for the segment level score. As a consequence, an expansion concept that dominates segments of the document that are highly relevant to the original query will be assigned a high likelihood.

We note that if we drop the dependence between adjacent segments from LCE_HMRF (i.e., we only consider a single segment S_j as evidence at a time), the model degrades into *segment-based LCE*, which we refer to as LCE_SB. With LCE_SB, documents are split into segments first, and then LCE is applied to segments for query expansion, treating each segment as a "document". This is akin to the passage-based query expansion performed by LCA [28]. While LCE_SB can exploit fine-grained document information, it cannot utilize valuable contextual information (e.g., information from sibling nodes). Thus, LCE_HMRF provides a generalization of passage-based query expansion approaches.

5. TIME COMPLEXITY

In this section, we compare the time complexity of LCE_GE and LCE_HMRF.

The efficiency of LCE_GE's joint distribution computation in Equation 3 is dominated by the clique sets dependent on the query, since clique sets that do not depend on the query can be precomputed at index-time. Among all the seven clique sets proposed for LCE_GE, the clique set U_{QE} is the most time-consuming one. For a query Q with $|Q|$ query terms, U_{QE} has $|Q|-1$ cliques (b, E, D) within it, where b is a query bigram. Given a fixed Q, the random variable E has $|V|$ (i.e., the size of the vocabulary) possible outcomes in total, and the random variable D has $|\mathcal{R}_{\mathcal{Q}}|$ (i.e., the number of feedback documents) possible outcomes. Thus, the feature function $f_{U_{QE}}(b, E, D)$ needs to be computed $(|Q|-1)\times|V|\times|\mathcal{R}_{\mathcal{Q}}|$ times. Supposing the index contains term position information and a term occurs tf_{avg} times within a document on average, then $f_{U_{QE}}(b, E, D)$ must be computed $O(tf_{avg}^3)$ times. As a result, the total complexity is $O(|Q|\cdot|V|\cdot|\mathcal{R}_{\mathcal{Q}}|\cdot tf_{avg}^3)$, where $|V|$, the vocab-

Name	Description	#Docs	Topics
Robust04	Robust 2004 data	528,155	301-450,601-700
WT10g	TREC Web data	1,692,096	451-550
GOV2	2004 crawl of .gov domain	25,205,179	701-850

Table 5: Summary of TREC data sets and topics.

ulary size, is often very large. Thus, LCE_GE suffers from high computational complexity problem.

Similarly, the efficiency of LCE_HMRF's joint distribution computation with Equation 4 is dominated by the computation of $f_{U_D}(b, D)$ and $f_{T_{SQ}}(q, S_{j-1}, S_j, D)$. We see that neither of these feature functions depends on the latent concept node E, and thus they do not need to be computed once for every term in the vocabulary. Hence, the computational complexity of the model is significantly decreased compared to LCE_GE. The time complexity of computing $f_{U_D}(b, D)$ is $O(|Q|\cdot|\mathcal{R}_\mathcal{Q}|\cdot tf_{avg}^2)$, while the time complexity of $f_{T_{SQ}}(q, S_{j-1}, S_j, D)$ is $O(|Q|\cdot|\mathcal{R}_\mathcal{Q}|\cdot tf_{avg}\cdot|S|_{avg})$ where $|S|_{avg}$ is the average number of segments per document.

6. EXPERIMENTS

In Section 3.2 and Section 4, we proposed two novel query expansion approaches called LCE_GE and LCE_HMRF for modeling the dependencies between query terms and expansion concepts. In this section, we evaluate the effectiveness of these approaches empirically. Experimental results show that these term dependencies can help improve the retrieval performance consistently and significantly during expansion.

6.1 Experimental Setup

Table 5 summarizes the data sets used in our experimental evaluation. The data sets vary by their document type (Robust04 is a newswire data set, while WT10g and GOV2 are web data sets), number of documents, and number of available topics, thus providing a diverse experimental setup for evaluating our proposed models.

All data sets were stopped using a standard list of 418 common terms and stemmed using a Porter stemmer. In all cases, only the title portions of the TREC topics are used in our experiments, to simulate short keyword queries. A modified version of the Indri search engine was used to index and rank documents [25].

Six different models are compared in our study, including: the unigram language modeling with Dirichlet smoothing (LM), the relevance model (RM3); latent concept expansion (LCE), the generalized LCE model (LCE_GE), LCE using Hierarchical Markov random fields (LCE_HMRF), and segment-based LCE (LCE_SB).

We compare LM, RM3, LCE, and LCE_HMRF to better understand how modeling term dependencies can contribute to retrieval performance during expansion. Additionally, we compare LCE_GE and LCE_HMRF to evaluate whether models that explicitly model dependencies between query terms and expansion terms are more effective than those that implicitly model such dependencies. Finally, we compare LCE_SB and LCE_HMRF to evaluate the usefulness of the segment-level interactions considered by LCE_HMRF compared to a simpler passage-based expansion approach.

We utilize cross-validation to estimate the parameters and evaluate the results for each data set. Given a data set, topics are divided into subsets of 50 topics each. Each subset, in turn, is used as a testing set, while the rest of the topics serve as a training set. Experiments are run separately for each data set, and average results over all testing sets are reported.

	LM	RM3	LCE	LCE_HMRF
Robust04	0.2532	0.2834^{α}	$0.3057^{\alpha\beta}$	$0.3313^{\alpha\beta\gamma}$
WT10g	0.1968	0.2118^{α}	$0.2259^{\alpha\beta}$	$0.2454^{\alpha\beta\gamma}$
GOV2	0.2981	0.3179^{α}	$0.3454^{\alpha\beta}$	$0.3634^{\alpha\beta\gamma}$

Table 6: Mean average precision for language modeling (LM), relevance model (RM3), latent concept expansion (LCE), and LCE using Hierarchical Markov random fields (LCE_HMRF). The superscripts α, β, and γ indicate statistically significant improvements ($p < 0.05$ using Wilcoxon test) over LM, RM3, LCE, respectively.

For the unigram language model, the smoothing parameter is trained. For RM3, LCE, LCE_GE, we train the model parameters, model hyperparameters, the number of pseudo-relevant documents used, and the number of expansion terms. For LCE_HMRF and LCE_SB, we also train the segment length. Meanwhile, LCE_HMRF adopts the same first-pass and second-pass retrieval algorithm (i.e. the MRF model for information retrieval [19]) as LCE, where the only difference between LCE_HMRF and LCE is how each selects and weights expansion terms.

6.2 Basic Results

In this section, we empirically evaluate the following hypothesis:

> Hypothesis **H1**: Query expansion approaches that properly consider term dependencies will perform better than approaches that do not consider all of dependencies essential for the query expansion task (i.e., the dependencies between query terms and the dependencies between query terms and expansion terms).

We note that the current state-of-the-art query expansion approaches usually improve retrieval effectiveness on average, but can also significantly hurt performance for individual queries unpredictably [8, 27, 10]. As a result, we measure our proposed approaches' effectiveness by two main criteria: average retrieval performance (in Section 6.2.1) and retrieval robustness (in Section 6.2.2).

6.2.1 Average Retrieval Performance

The mean average precision for LM, RM3, LCE, and LCE_HMRF are given in Table 6. As would be expected, the relevance model, LCE, and LCE_HMRF always significantly outperform the unigram language model.

Furthermore, LCE significantly outperforms the relevance model across all data sets. This indicates that term dependencies can provide extra information to help estimate a better query model than using independent terms alone and can contribute to the retrieval performance during expansion. The findings agree with similar experiments that were previously carried out [20].

As the results show, by implicitly incorporating term dependencies between query terms and expansion terms, LCE_

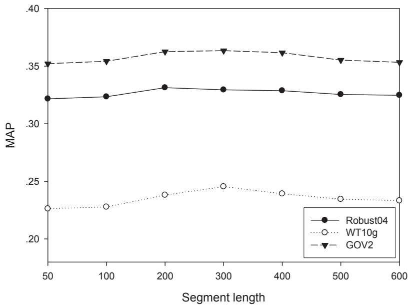

Figure 4: Sensitivity to segment length.

HMRF achieves the best performance across all of the data sets. In particular, it achieves 16.9%, 15.9%, and 14.3% relative improvements in mean average precision over the relevance model, on Robust04, WT10g, and GOV2, respectively. Moreover, LCE_HMRF shows significant improvements over the original LCE approach, which demonstrates the benefits of using the document structure to implicitly model dependencies between query terms and expansion terms during expansion and support hypothesis **H1**. The relative improvements over LCE are 8.4% for Robust04, 8.6% for WT10g, and 5.2% for GOV2. All of the improvements, with respect to relevance models and LCE are statistically significant.

6.2.2 Retrieval Robustness

As we have observed, the relevance model, LCE, and LCE_HMRF can lead to significantly improvements in retrieval effectiveness on average versus a simple unigram language modeling baseline. Here, we demonstrate and compare the robustness of these query expansion techniques with respect to this baseline. We define robustness as the number of queries whose effectiveness is improved/degraded (and by how much) as the results of applying these methods. A highly robust expansion technique will significantly improve many queries and only minimally degrade very few.

Figure 3 presents an analysis of the robustness of the relevance model, LCE, and LCE_HMRF on Robust04, WT10g, and GOV2. The histograms present, for various ranges of relative decreases/increases in mean average precision, the number of queries that are hurt/improved over the baseline unigram language model.

The models that consider term dependencies show strong robustness for each data set. For the Robust04 data set, the relevance model improves 182 queries and degrades 67, whereas LCE improves 190 and degrades 59 and LCE_HMRF improves 188 and degrades 61. Although LCE improves the effectiveness of 2 more queries than LCE_HMRF, the relative improvement exhibited by LCE_HMRF is significantly larger. For WT10g, the relevance model improves 52 and degrades 47, while LCE improves 61 and degrades 38 and LCE_HMRF improves 66 and degrades 33. Finally, for GOV2, the relevance model improves 92 and degrades 56, while LCE improves 104 and degrades 44 and LCE_HMRF improves 116 and degrades 32. Therefore LCE_HMRF exhibits good robustness characteristics.

6.2.3 Parameter Sensitivity

Finally, we note that our proposed HMRF model relies on automatically breaking documents into segments. Therefore, we are interested in analyzing the sensitivity of the retrieval performance with respect to the size of the segment. This sensitivity analysis is shown in Figure 4. For each segment length, we train all of the other model parameters to fairly evaluate the sensitivity. The results show that the effectiveness is relatively stable across different segment lengths. However, for Robust04, the homogeneous newswire data set, *segment length=200* performed the best. For the heterogeneous web data sets such as WT10g and GOV2, *segment length=300* performed the best. These results suggest that setting the segment length in the range of 200-300 is a reasonable 'default' setting, at least for the data sets currently under consideration.

	LCE	LCE_GE	LCE_HMRF
Robust04	0.3057	0.3092	$0.3313^{\alpha\beta}$
WT10g	0.2259	0.2287	$0.2454^{\alpha\beta}$

Table 7: Mean average precision for LCE, LCE_GE, LCE_HMRF. The superscripts α and β indicate statistically significant improvements ($p < 0.05$ using Wilcoxon test) over LCE and LCE_GE, respectively.

6.3 LCE_GE vs. LCE_HMRF

In this section, we empirically evaluate the following hypothesis:

> Hypothesis **H2**: Hierarchical models that seamlessly integrate the hierarchical document structure can address the high computational complexity and data sparseness problems for modeling term dependencies, which are suffered by non-hierarchical models.

We compare the effectiveness of LCE_GE, which is an non-hierarchical model and explicitly models the dependencies between query terms and expansion concepts, and LCE_HMRF, which models the same type of dependencies implicitly via the hierarchical document structure.

The results of this comparison are provided in Table 7. The results show that LCE_HMRF significantly outperforms LCE_GE on all data sets. The GOV2 data set is not tested due to the high computational cost involved in applying LCE_GE. Interestingly, LCE_GE shows improvements over LCE, but the improvements are not statistically significant. Although LCE_GE and LCE_HMRF both model dependencies between query terms and expansion concepts, LCE_GE is shown to not be as effective as LCE_HMRF, presumably due to the data sparseness issue described earlier.

Additionally, the computational cost of LCE_GE is much higher than LCE_HMRF. Unlike LCE_HMRF, LCE_GE explicitly models a large number of pairs of query terms and latent concepts. Thus, LCE_GE has to compute many term proximity features over these pairs, which is very time consuming. Therefore, LCE_HMRF is a practically appealing query expansion approach, both in terms of efficiency and effectiveness. Of course, in a real, large-scale system, query expansion would likely be either done offline or done online using a hybrid strategy, such as the one described by Broder et al. [4]. However, even when such an architecture is used, LCE_HMRF is still desirable to LCE_GE. As a consequence, we can support hypothesis **H2**.

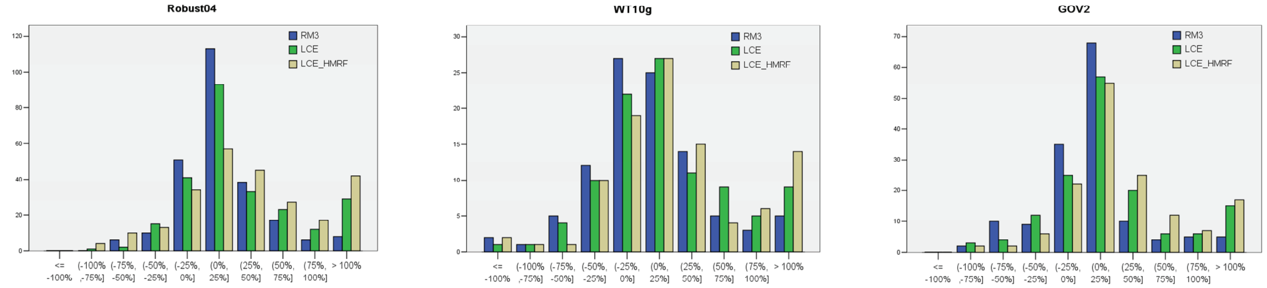

Figure 3: Histograms that compare the robustness of the relevance model (RM3), latent concept expansion (LCE), and LCE using Hierarchical Markov random fields (LCE_HMRF) with respect to the unigram language model (LM) for the Robust04, WT10g, and GOV2 data sets.

	LCE	LCE_SB	LCE_HMRF
Robust04	0.3057	0.2981	$0.3313^{\alpha\beta}$
WT10g	0.2259	0.2182	$0.2454^{\alpha\beta}$
GOV2	0.3454	0.3511	$0.3634^{\alpha\beta}$

Table 8: Mean average precision for LCE, LCE_SB, and LCE_HMRF. The superscripts α and β indicate statistically significant improvements ($p < 0.05$ using Wilcoxon test) over LCE and LCE_SB, respectively.

	LM	MRF	Ranking_HMRF
Robust04	0.2532	0.2653^{α}	$0.2684^{\alpha\beta}$
WT10g	0.1968	0.2074^{α}	$0.2144^{\alpha\beta}$
GOV2	0.2981	0.3244^{α}	$0.3332^{\alpha\beta}$

Table 9: Mean average precision for LM, MRF, and Ranking_HMRF. The superscripts α and β indicate statistically significant improvements ($p < 0.05$ using Wilcoxon test) over LM and MRF, respectively.

6.4 Segment-Based LCE vs. LCE_HMRF

As mentioned earlier, the dependencies between sibling nodes and the parent node considered by the HMRF model provides a mechanism for utilizing valuable contextual information inside a document. To quantify the utility of these dependencies, we compare the effectiveness of LCE_SB (passage-based LCE) and LCE_HMRF.

As shown in Table 8, LCE_HMRF always significantly outperforms LCE_SB. In fact, LCE_SB shows slightly lower performance than LCE on the Robust04 and WT10g data sets. Although LCE_SB has the capability to exploit a document's fine-grained evidence, it fails to significantly improve the retrieval performance over LCE, likely due to data sparseness, which stems from the fact that LCE_SB only considers information from a single segment, instead of from two adjacent segments and the entire document, as is the case in LCE_HMRF. The segment-level dependencies considered by LCE_HMRF can help alleviate this problem by leveraging the document's structure information to smooth the segments and improve the retrieval performance. Therefore, LCE_HMRF is more than simply a passage-based version of LCE, as these results clearly show that modeling intra-segment dependencies is highly effective.

6.5 HMRFs for Ranking

Our proposed novel graphical model called HMRFs provide a natural, formally motivated mechanism for modeling term dependencies and document structure. Thus, we are interested in exploring how effective the model is for ranking compared to the original MRF model [19], independent of the query expansion task. As illustrated in Figure 2, by removing the latent concept node from the model structure, the HMRF model defines a joint distribution over a query and a triplet of nodes in the document tree. Hence, the conditional probability of a triplet given the query can easily be computed and utilized for ranking. When HMRFs are used in this way (i.e., directly for ranking document instead of expanding queries), we refer to the resulting approach as Ranking_HMRF. In this work, we simply use the best match strategy to score a document for ranking, as follows:

$$\begin{aligned} Score(Q,D) &= max\{P(S_{j-1},S_j,D|Q)\} \\ &\overset{rank}{=} max\{P(Q,S_{j-1},S_j,D)\} \end{aligned} \tag{6}$$

where $\overset{rank}{=}$ denotes rank equivalence, $\{S_{j-1},S_j,D\}$ is the set of triplets extracted from document D, and $P(Q,S_{j-1},S_j,D)$ is the joint distribution of Ranking_HMRF.

We compare the effectiveness of Ranking_HMRF with the original MRF model in Table 9. Both MRF and Ranking_HMRF employ the sequential dependency assumption, which assumes that dependencies exist between adjacent query terms. For efficiency reasons, we conduct the Ranking_HMRF approach by reranking the top 3000 results returned by the baseline approach (i.e., unigram language model [23]).

As shown in the table, Ranking_HMRF consistently and significantly outperforms the MRF model across all the data sets. We attribute the additional performance gain to the hierarchical document structure which is naturally captured by Ranking_HMRF. Basically, the original MRF model can capture two kinds of features, namely term occurrence and term proximity features. The former features are loosely defined at the document level while the latter ones are strictly defined within a limited window size (e.g., 8 terms). The document structure aware Ranking_HMRF approach provides a way for exploring co-occurrence at different spatial scales, and can define features at segment level which are less strict than the proximity features and more focused than the term occurrence features.

7. CONCLUSION AND FUTURE WORK

In this paper, we showed that most previously proposed query expansion approaches do not properly model dependencies between query terms and expansion concepts and make overly simple assumptions about document structure. As a result, we proposed a novel query expansion paradigm, built upon LCE, that models the dependencies between query terms and expansion terms. We have shown that these dependencies can help improve the retrieval performance during expansion. To effectively and efficiently model these important term dependencies, we also introduced a model called Hierarchical Markov random fields (HMRFs). We have demonstrated, both theoretically and experimentally, that LCE using HMRFs can (partially) avoid the high computational complexity and the data sparseness issues that plague other models. In particular, by modeling dependencies between adjacent document segments and the entire document, HMRFs provide a mechanism to utilize a document's structural information to smooth the segments within it. Experimental results showed that LCE_HMRF provides state-of-the-art query expansion retrieval effectiveness across several TREC data sets.

Our work can be extended in several ways. First, only term occurrence and term proximity features are used when we design the feature functions. There is additional information that can be used, such as named entities, text style, PageRank, and readability, among others. Such information can also be considered in our models. Second, our work has focused on dealing with unstructured data only. In the Web environment, there exists large amount of structured and semi-structured data containing HTML or XML fields that often play an important role in retrieval. It would be interesting to see how LCE_HMRF can be extended to deal with these kinds of data. Third, it would be interesting to incorporate some notion of risk directly into the model, as was recently proposed by Collins-Thompson [6, 7].

8. ACKNOWLEDGMENTS

This work is supported by the Major State Basic Research Project of China under Grant No. 2007CB311103, the National High Technology Research and Development Program of China under Grant No. 2006AA010105, and the National Science Foundation of China under Grant No. 60776797 and 60873166.

9. REFERENCES

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade. Umass at trec 2004: Novelty and hard. In *Proceedings of TREC '04*, 2004.

[2] M. Bendersky, D. Metzler, and W. B. Croft. Learning concept importance using a weighted dependence model. In *Proceedings of WSDM '10*, pages 31–40, 2010.

[3] D. M. Blei and P. J. Moreno. Topic segmentation with an aspect hidden markov model. In *Proceedings of SIGIR '01*, pages 343–348, 2001.

[4] A. Broder, P. Ciccolo, E. Gabrilovich, V. Josifovski, D. Metzler, L. Riedel, and J. Yuan. Online expansion of rare queries for sponsored search. In *Proceedings of WWW '09*, pages 511–520, 2009.

[5] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of SIGIR '08*, pages 243–250, 2008.

[6] K. Collins-Thompson. Estimating robust query models with convex optimization. In *Proceedings of NIPS '08*, pages 329–336, 2008.

[7] K. Collins-Thompson. Reducing the risk of query expansion via robust constrained optimization. In *Proceedings of CIKM '09*, pages 837–846, 2009.

[8] D. Harman and C. Buckley. The nrrc reliable information access (ria) workshop. In *Proceedings of SIGIR '04*, pages 528–529, 2004.

[9] B. He and I. Ounis. Finding good feedback documents. In *Proceedings of CIKM '09*, pages 2011–2014, 2009.

[10] B. He and I. Ounis. Studying query expansion effectiveness. In *Proceedings of ECIR '09*, pages 611–619, 2009.

[11] X. Ji and H. Zha. Domain-independent text segmentation using anisotropic diffusion and dynamic programming. In *Proceedings of SIGIR '03*, pages 322–329, 2003.

[12] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of KDD '02*, pages 133–142, 2002.

[13] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of SIGIR '01*, pages 120–127, 2001.

[14] K. S. Lee, W. B. Croft, and J. Allan. A cluster-based resampling method for pseudo-relevance feedback. In *Proceedings of SIGIR '08*, pages 235–242, 2008.

[15] T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends in Inf. Retr.*, 3:225–331, 2009.

[16] Y. Lv and C. Zhai. Positional language models for information retrieval. In *Proceedings of SIGIR '09*, pages 299–306, 2009.

[17] C. Macdonald and I. Ounis. Expertise drift and query expansion in expert search. In *Proceedings of CIKM '07*, pages 341–350, 2007.

[18] D. Metzler and W. Bruce Croft. Linear feature-based models for information retrieval. *Inf. Retr.*, 10(3):257–274, 2007.

[19] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *Proceedings of SIGIR '05*, pages 472–479, 2005.

[20] D. Metzler and W. B. Croft. Latent concept expansion using markov random fields. In *Proceedings of SIGIR '07*, pages 311–318, 2007.

[21] W. Morgan, W. Greiff, and J. Henderson. Direct maximization of average precision by hill-climbing, with a comparison to a maximum entropy approach. In *Proceedings of HLT-NAACL '04*, pages 93–96, 2004.

[22] V. Murdock and W. B. Croft. A translation model for sentence retrieval. In *Proceedings of HLT '05*, pages 684–691, 2005.

[23] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR '98*, pages 275–281, 1998.

[24] J. J. Rocchio. *Relevance Feedback in Information Retrieval*. Prentice-Hall, 1971.

[25] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language model-based serach engine for complex queries. In *Proceedings of the International Conference on Intelligence Analysis*, 2004.

[26] B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In *Proceedings of NIPS '03*, 2003.

[27] E. M. Voorhees. Overview of the trec 2004 robust retrieval track. In *Proceedings of TREC '04*, 2004.

[28] J. Xu and W. B. Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18:79–112, 2000.

[29] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *Proceedings of SIGIR '07*, pages 271–278, 2007.

[30] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of CIKM '01*, pages 403–410, 2001.