Effective Query Formulation with Multiple Information Sources

Michael Bendersky
Dept. of Computer Science
U. of Massachusetts
Amherst, MA
bemike@cs.umass.edu

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

W. Bruce Croft
Dept. of Computer Science
U. of Massachusetts
Amherst, MA
croft@cs.umass.edu

ABSTRACT

Most standard information retrieval models use a single source of information (e.g., the retrieval corpus) for query formulation tasks such as term and phrase weighting and query expansion. In contrast, in this paper, we present a unified framework that automatically optimizes the combination of information sources used for effective query formulation. The proposed framework produces fully weighted and expanded queries that are both more effective and more compact than those produced by the current state-of-the-art query expansion and weighting methods. We conduct an empirical evaluation of our framework for both newswire and web corpora. In all cases, our combination of multiple information sources for query formulation is found to be more effective than using any single source. The proposed query formulations are especially advantageous for large scale web corpora, where they also reduce the number of terms required for effective query expansion, and improve the diversity of the retrieved results.

Categories and Subject Descriptors

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

General Terms

Algorithms, Experimentation

Keywords

Query formulation, query expansion, external sources

1. INTRODUCTION

Most of today's commercial web search engines heavily rely on sophisticated machine learned ranking functions to produce high quality results. Such ranking functions typically combine evidence from hundreds, or even thousands, of different features[1]. Amongst the most basic and essential of these features are those that match the text of the query to the text of the document, which are often referred to as *text matching features*. The quality of text matching features can often be a key factor in the success of the learned ranking function [21].

The derivation of text matching features depends on the process of *query formulation* – a process during which the original keyword query issued by the user is transformed into a structured query representation that is consumed by the search engine. A standard query formulation process usually involves query segmentation into atomic concepts, weighting of these concepts and query expansion with related concepts, among other possible transformations.

Up until now, most of the research that has gone into improving the process of query formulation has been fragmented, and has not produced a unified query formulation framework. Without such a framework, it is difficult for researchers and practitioners to systematically and monotonically improve the effectiveness of the text matching capabilities of their retrieval systems. Instead, to achieve a high level of effectiveness, it is often necessary to mix and match ideas from multiple, competing methodologies or techniques in ways that are often heuristic, inefficient, or sub-optimal.

Nearly all of the recent advances in query formulation have been a result of research on improving one of the following types of query transformation: identification and matching of atomic query concepts, concept weighting and query expansion. For effective concept identification and matching, researchers employed retrieval methods based on the Markov random field (MRF) model [6, 28] and the positional language models [24, 25]. For improved term weighting, researchers proposed unsupervised methods such as term frequency saturation, document length normalization, and field weighting (e.g., BM25F [34]), as well as supervised methods that estimate the global term importance based on a variety of external sources (e.g., number of times a term occurs in a query log or a title of a Wikipedia article) [20, 19, 6, 7]. Finally, best practice query expansion techniques include, among others, positional relevance models [25], latent concept expansion (LCE) [29], parameterized query expansion (PQE) [7], and expansion using external corpora [12, 45].

While there has been some synergistic research across these types of query transformations (e.g., positional language models giving rise to positional relevance models [25]),

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

[1]For instance, a recently released dataset by Yahoo! (`http://learningtorankchallenge.yahoo.com/`) includes approximately 700 features.

Latent Concept Expansion *(Retrieval Corpus)*

Query	Expansion Terms
0.479 er	0.145 tv
0.479 tv	0.112 er
0.479 show	0.055 folge
0.120 er tv	0.054 selbst
0.120 tv show	0.034 show
	...

Mean Average Precision 12.29

Latent Concept Expansion *(Wikipedia)*

Query	Expansion Terms
0.464 er	0.156 tv
0.464 tv	0.074 bisexual
0.464 show	0.066 film
0.116 er tv	0.064 season
0.116 tv show	0.059 series
	...

Mean Average Precision 25.68

Query Formulation *(Multiple Sources)*

Query	Expansion Terms
0.297 er	0.085 season
0.168 tv	0.065 episode
0.192 show	0.051 dr
0.051 er tv	0.043 drama
0.012 tv show	0.036 series
	...

Mean Average Precision 38.31

Table 1: Comparison of the performance of the latent concept expansion with retrieval corpus or Wikipedia to the performance of the query formulation using multiple information sources for the query *"ER TV Show"*.

there does not exist a robust, unified framework capable of encompassing the current state-of-the-art across all transformation types. Accordingly, in this paper, we propose a novel *query formulation framework* that combines the best aspects of each of these transformations in a highly robust and effective manner and produces a richly structured representation of a keyword query.

As an example, Table 1 compares the output of the proposed query formulation method for the keyword query *"ER TV Show"* to the output of the latent concept expansion (LCE) method [29] that uses either the retrieval corpus or the Wikipedia corpus for query expansion. Both of these expansion strategies have been shown to be highly effective strategies in previous work [29, 16, 26, 45]. It is clear from Table 1 that there are two main advantages of our query formulation approach compared to the LCE method.

First, LCE assumes equal importance among query terms and query phrases by assigning them fixed weights. On the other hand, our query formulation assigns relative importance weights, based on evidence from multiple information sources, to explicit query terms and phrases. For instance, in the context of the query *"ER TV Show"*, the most important term is *"er"* and the phrase *"er tv"* is more important than the phrase *"tv show"*.

Second, LCE uses a single source for expansion, which can sometimes lead to topic drift. As a case in point, in Table 1, LCE with the retrieval corpus expands the query with non-English terms *folge* and *selbst*, and LCE with Wikipedia expands the query with non-helpful terms *bisexual* and *film*. To combat topic drift, the proposed query formulation method combines evidence from multiple sources (including, among others, the retrieval corpus and Wikipedia) to derive a relevant and diverse list of expansion terms.

Due to these advantages, we hypothesize that a unified query formulation approach that uses multiple information sources will yield better results than any of the previously proposed query transformation methods in isolation. In fact, for the query in Table 1, our query formulation improves the retrieval performance by 50% compared to the best performing LCE-based method.

The query formulation method presented in this paper synthesizes three main research directions. First, it incorporates the highly effective term proximity matching of the sequential dependence model, which was first proposed by Metzler and Croft [28]. Second, it incorporates the state-of-the-art parameterized concept weighting framework recently proposed by Bendersky et al. [7]. Finally, it is inspired by previous work that demonstrates that query expansion using external corpora is highly effective [12, 23, 45].

The end result of this synthesis is a unified framework, which distills effective and compact query formulations, such as the one shown in Table 1. Empirical results show that these query formulations are significantly more effective than many of the current state-of-the-art text matching methods used as baselines.

This work has three primary contributions. First, we develop a novel unified query formulation framework that (a) supports arbitrary query concepts (e.g., unigrams, bigrams, expansion terms, etc.); (b) supports explicit query concept weighting; and (c) extracts and weights expansion terms using a parameterized approach that leverages evidence from external information sources. All of the weighting and expansion models are learned using a simple, yet effective learning to rank approach.

Second, the proposed query formulation framework naturally gives rise to a text matching function that scores a query formulation with respect to a document. This matching function can be used alone as a (text-only) retrieval model or within a machine learned ranking function as a highly effective text matching feature.

Finally, we carry out a detailed experimental evaluation of the proposed framework. Our results demonstrate that the proposed approach achieves state-of-the-art effectiveness compared to a number of highly competitive baseline systems. Our experimental evaluation also shows that our proposed approach has desirable efficiency and robustness properties, and also achieves strong performance in terms of a number of diversity metrics.

The remainder of this paper is laid out as follows. First, Section 2 describes the theoretical underpinnings of the proposed query formulation framework. Next, Section 3 discusses the external sources of evidence that we use to formulate highly effective queries. Section 4 highlights related work, while Section 5 presents the findings of our experimental evaluation. Finally, Section 6 concludes the paper.

2. QUERY FORMULATION

The process of *query formulation* (also referred to as *query rewriting* or *query transformation* [11]) modifies the original keyword query submitted by the user to the search engine in order to better represent the underlying intent of the query. The formulated query is then used as an input to the search engine's ranking algorithm. Thus, the primary goal of query formulation is to improve the overall quality of the ranking presented to the user in response to her query [11].

Query formulation is usually divided into two main processing stages. The first processing stage, which is usually referred to as *query refinement* [13], alters the query on the

morphological level (e.g., tokenization, spelling corrections, stemming, etc.).

After the query refinement stage is completed, the second processing stage alters the query on the structural level. Such structural alterations may include, among other actions, segmenting the query into atomic concepts (i.e., combinations of terms), assigning weights to these concepts, or expanding the query with related weighted concepts.

In this work, our focus is on the structural stage of query formulation. Hence, we assume that all the input queries are either spelled and tokenized correctly, or have undergone the query refinement process.

Accordingly, given an input keyword query Q, we assume that we can identify a weighted set of concepts $\mathcal{K}_Q$, which can be associated with the user intent underlying this query. Note that the concepts in the set $\mathcal{K}_Q$ can be either explicitly present in the query Q, or associated with it via some process of query expansion (e.g., pseudo-relevance feedback [44]).

Once the set of concepts $\mathcal{K}_Q$ is identified, and the concept weights are determined, we can score the documents in the collection using a linear weighted combination of the matches of concepts in the set $\mathcal{K}_Q$, and rank the documents based on this score. Formally, the score of document D in the collection can be written as

$$sc(Q, D) \triangleq \sum_{\kappa \in \mathcal{K}_Q} \lambda(\kappa, Q) f(\kappa, D) \quad (1)$$

The ranking function in Equation 1 consists of two components. First, a concept matching function $f(\kappa, D)$ measures the relatedness between the document D and the concept κ. Second, a concept weighting function $\lambda(\kappa, Q)$ measures the importance of the concept κ for query Q. Intuitively, Equation 1 assigns higher scores to documents that match more of the important concepts related to the query. In the remainder of this section, we detail the modeling and the estimation of the concept matching function and the concept weighting function.

2.1 Concept Matching Function

The concept matching function $f(\kappa, D)$ assigns a score to the matches of concept κ in the document D. This function may take various forms, however in information retrieval applications it is commonly a monotonic function, i.e., its value increases with the number of times concept κ matches document D.

In this paper, we assume that the matching function $f(\kappa,D)$ is estimated using the log of the probability of concept κ given document D with Dirichlet smoothing [46], i.e.,

$$f(\kappa, D) = \log \frac{tf_{\kappa,D} + \mu \frac{tf_{\kappa,\mathcal{C}}}{|\mathcal{C}|}}{|D| + \mu}, \quad (2)$$

where $tf_{\kappa,D}$ and $tf_{\kappa,\mathcal{C}}$ are the number of concept occurrences in the document and the collection, respectively; μ is a free parameter; $|D|$ is the number of terms in D, and $|\mathcal{C}|$ is the total number of terms in the collection.

We use this probabilistic estimate as a concept matching function since it is convenient and efficient to compute and exhibits state-of-the-art retrieval performance in other concept-based retrieval models [7, 28, 29]. However, other commonly used matching functions (such as BM25 [35] or DFR [2]) can be substituted in Equation 1 without loss of generality.

2.2 Concept Weighting Function

The concept importance function $\lambda(\kappa, Q)$ measures the importance of concept κ for conveying the user intent underlying the query Q. In its simplest form, the concept importance function may be a single collection statistic associated with the concept κ such as inverse document frequency [40].

Recently, researchers have found that supervised models of concept weighting that leverage statistics from external information sources (e.g., query logs, Wikipedia, large n-gram repositories, large newswire collections, etc.) can significantly improve the retrieval performance [20, 19, 6, 7]. Thus far, however, these models were mainly used for weighting the explicit query concepts [20, 6] or re-weighting the expansion terms that were associated with the query via pseudo-relevance feedback using the retrieval corpus [7, 8].

In contrast, in this section we show that external information sources can also be used, in addition to concept weighting, to select and weight related and helpful terms with which the original query can be expanded. As the example in Table 1 demonstrates, such terms can be more relevant and diverse than the expansion terms that are obtained through the standard process of pseudo-relevance feedback on the retrieval corpus [17, 29].

To this end, we define a set of external information sources $\mathcal{S}$, which we use as a basis for deriving features for either concept weighting or query expansion. To make our approach as widely applicable as possible, we make no assumptions about the internal structure of these sources, and treat them as standard unstructured textual corpora.

In Section 2.2.1 we explain how to use the set of external sources $\mathcal{S}$ for weighting the explicit query concepts. Then, in Section 2.2.2, we describe how the set of external sources $\mathcal{S}$ is used to expand the original query with new related terms. We defer the precise definition of the external information sources in the set $\mathcal{S}$ used for weighting and expansion to Section 3.

2.2.1 Explicit Query Concepts

Following previous work, we define an explicit query concept as any combination of terms that appears in the query and can be matched within a document in the retrieval corpus [6, 7]. In particular, we use a subset of concepts first proposed by Metzler and Croft [28], and restrict our attention to single terms and adjacent bigram phrases and proximities. This provides a set of explicit query concepts, which we refer to as $\mathcal{X}_Q$. The set of concepts $\mathcal{X}_Q$ is compact (it is linear in the number of query terms), and has been shown to attain highly effective retrieval performance [6, 7, 28, 29, 16].

To weight a concept $\kappa_x \in \mathcal{X}_Q$, we use the *parameterized weighting* approach, proposed by Bendersky et al. [6, 7], which leverages the statistics from external information sources to assign reliable weights to the concepts in the query. Formally, the concept weight of κ_x is modeled as

$$\lambda(\kappa_x, Q) = \sum_{\varphi \in \Phi_{\mathcal{S}}} w_\varphi \varphi(\kappa_x, Q, \sigma). \quad (3)$$

As can be seen from Equation 3, the weight $\lambda(\kappa_x, Q)$ of the concept $\kappa_x \in \mathcal{X}_Q$ is expressed by a weighted combination of *importance features* $\Phi_{\mathcal{S}}$, which are defined over a set of sources $\mathcal{S}$. Each importance feature $\varphi \in \Phi_{\mathcal{S}}$ is associated with an explicit query concept κ_x and is computed over a source $\sigma \in \mathcal{S}$.

2.2.2 Expansion Terms

A key observation from Equation 1 is that the proposed ranking function is not limited to the set of explicit query concepts $\mathcal{X}_Q$ defined in the previous section. Instead, the ranking function may include expansion concepts from sources other than the search query or the retrieval corpus, which has been the standard practice in previous work [6, 7, 8, 28, 29]. While any combination of terms can serve as an expansion concept, in this work we focus on expansion with single terms, mainly for ensuring the efficiency of the expansion concept selection process.

To incorporate expansion terms from external sources in the set $\mathcal{S}$, we first obtain a large pool of potential expansion terms associated with an information source $\sigma \in \mathcal{S}$ using pseudo-relevance feedback. To this end, we first rank documents in the source σ using the ranking function

$$sc(Q, D) \triangleq \sum_{\kappa_x \in \mathcal{X}_Q} \lambda(\kappa_x, Q) f(\kappa_x, D), \tag{4}$$

which utilizes only explicit query concepts and their corresponding weights. Then, each term in the pseudo-relevant set of documents $\mathcal{R}_\sigma$ (top ranked documents in source σ) is assigned an *expansion score*

$$\psi(\kappa, Q, \sigma) = \sum_{D \in \mathcal{R}_\sigma} \exp\Big(\gamma_1 sc(Q, D) + \gamma_2 f(\kappa, D) - \gamma_3 \log \frac{tf_{\kappa,\sigma}}{|\sigma|}\Big), \tag{5}$$

where γ_i's are free parameters.

Note that Equation 5 uses the same concept expansion weighting scheme used by Latent Concept Expansion [29]. The score $\psi(\kappa, Q, \sigma)$ is a linear combination of three key components: document relevance (manifested by the document score $sc(Q,D)$), weight of the term in the pseudo-relevant set $\mathcal{R}_\sigma$ (manifested by the matching function $f(\kappa,D)$), and the inverse of the frequency of the term in the source σ ($-\log \frac{tf_{\kappa,\sigma}}{|\sigma|}$), which dampens the scores of very common terms, thereby improving the quality of the initial pool $\mathcal{E}_Q^{init}$.

Finally, at most M terms with the highest value of $\psi(\kappa, Q, \sigma)$ per source σ are added to $\mathcal{E}_Q^{init}$, the initial pool of expansion terms[2]. This ensures that the size of the initial pool $\mathcal{E}_Q^{init}$ is bounded by $|\mathcal{E}_Q^{init}| \leq M|\mathcal{S}|$.

Once the initial pool $\mathcal{E}_Q^{init}$ is obtained, we assign a weight to each unique term $\kappa_e \in \mathcal{E}_Q^{init}$, using the weighted combination of expansion scores

$$\lambda(\kappa_e, Q) = \sum_{\psi \in \Psi_\mathcal{S}} w_\psi \psi(\kappa_e, Q, \sigma). \tag{6}$$

According to Equation 6, the weight $\lambda(\kappa_e, Q)$ of the term $\kappa_e \in \mathcal{E}_Q^{init}$ is expressed by a weighted combination of *expansion scores* $\Psi_\mathcal{S}$, which is defined over a set of sources $\mathcal{S}$. Each expansion score $\psi \in \Psi_\mathcal{S}$ is associated with an expansion term κ_e and query Q, and is computed over a source $\sigma \in \mathcal{S}$. To handle missing terms, if κ_e is not one of the top M terms selected per source σ, we set $\psi(\kappa_e, Q, \sigma) = 0$.

To ensure efficient query expansion, we retain only the top K terms[3] from the initial pool $\mathcal{E}_Q^{init}$, based on Equation 6. We refer to this small set of expansion terms as $\mathcal{E}_Q$.

[2]To ensure that the initial pool $\mathcal{E}_Q^{init}$ is large enough for selecting diverse expansion terms, we set $M = 100$.

[3]In all the experiments in this paper, $K \leq 10$.

2.3 Parameter Estimation

To better illustrate the free parameters that need to be estimated in order to complete the query formulation derivation, we substitute the weighting function $\lambda(\kappa, Q)$ in Equation 1 with its derivations in Equation 3 and Equation 6, which yields

$$\begin{aligned} sc(Q, D) \triangleq \sum_{\kappa \in \mathcal{K}_Q} \lambda(\kappa, Q) f(\kappa, D) = \\ \sum_{\varphi \in \Phi_\mathcal{S}} w_\varphi \sum_{\kappa_x \in \mathcal{X}_Q} \varphi(\kappa_x, Q, \sigma) f(\kappa_x, D) + \\ \sum_{\psi \in \Psi_\mathcal{S}} w_\psi \sum_{\kappa_e \in \mathcal{E}_Q} \psi(\kappa_e, Q, \sigma) f(\kappa_e, D). \end{aligned} \tag{7}$$

Equation 7 demonstrates that the final document score is determined by the combination of explicit query concept matches (matches in the set $\mathcal{X}_Q$), and the expansion term matches (matches in the set $\mathcal{E}_Q$). Accordingly, the free parameters in Equation 7 are: (a) the parameters that combine the importance features (w_φ), and (b) the parameters that combine the expansion scores (w_ψ).

To estimate these two sets of parameters, we use a three-stage optimization approach, which leverages the available relevance data as a training set. We use this three-stage optimization approach due to the fact that the setting of the parameters w_φ will influence the choice of the set of expansion terms in the initial set $\mathcal{E}_Q^{init}$, which will consequently influence the setting of the parameters w_ψ (as described in Section 2.2.2) that control the choice of terms in the expansion set $\mathcal{E}_Q$.

Stage I First, we optimize the set of parameters that combine the importance features

$$\{w_\varphi | \varphi \in \Phi_\mathcal{S}\}.$$

This determines the weights assigned to the explicit query concepts in the set $\mathcal{X}_Q$ (see Equation 3).

Stage II We then obtain an initial set of expansion terms $\mathcal{E}_Q^{init}$, by using the parameters w_φ (obtained at Stage I) in Equation 4.

Stage III Finally, we optimize the set of parameters that combine the expansion scores

$$\{w_\psi | \psi \in \Psi_\mathcal{S}\}.$$

This determines the choice of top K expansion terms in the set $\mathcal{E}_Q$ used for query formulation and their weights (see Equation 6).

For parameter optimization at Stage I and Stage III, we use the coordinate ascent (CA) algorithm proposed by Metzler and Croft [30]. The CA algorithm iteratively optimizes a target metric on a given training set of query-document relevance judgments (in our case, retrieval metric such as MAP) by performing a series of one-dimensional line searches. It repeatedly cycles through each of the parameters w, holding all other parameters fixed while optimizing it. This process is performed iteratively over all parameters until the gain in the target metric is below a certain threshold. Although we use the CA algorithm primarily for its simplicity, efficiency and effectiveness, any other learning to rank approach that optimizes the parameters for linear models (see Li [21] for a survey on learning to rank techniques) can be adopted as well.

Sources used for concept weighting		Sources used for query expansion	
Source	*Extracted Importance Features*	*Source*	*Unit of Retrieval*
`Google N-grams`	Frequency of concept κ	`ClueWeb Heading Text`	Single line of heading text (as defined by the <h1> – <h6> tags)
`MSN Query Log`	Frequency of concept κ		
`Wikipedia Titles`	Frequency of concept κ	`ClueWeb Anchor Text`	Single line of anchor text (as defined by the <a> tag)
`Retrieval Corpus`	Document frequency of concept κ		
	Collection frequency of concept κ	`Wikipedia Corpus`	Single article
		`Retrieval Corpus`	Single document

Table 2: External information sources used for concept importance weighting (Section 2.2.1) and query expansion (Section 2.2.2). All the frequency features are log-scaled.

`Retrieval Corpus`	`Wikipedia`	`Anchor Text`	`Heading Text`	***Combined***
chemical, weapon, toxic, convention, substance, gas, destruction, product, plant, mirzayanov, ...	chemical, agent, gas, weapon, warfare, war, poison, mustard, disseminate, nerve, ...	toxic, chemical, cigarette, tobacco, terrorist, tts, weapon, leach, terror, wwf, ...	toxic, chemical, weapon, terrorist, terror, assess, biology, behavior, incinerate, emission, ...	*weapon*, *agent*, *gas*, *russia*, *convention*, *mustard*, *warfare*, *substance*, *destruction*, *product*, ...

Table 3: Comparison between the lists of expansion terms derived from the individual external information sources for the query *"toxic chemical weapon"* and the combined list produced by our query formulation method.

3. INFORMATION SOURCES

In this section, we provide a detailed description of the set of external information sources $\mathcal{S}$ used for query formulation. As described in Section 2.2, we make no assumptions about the internal structure of these sources, and treat them as unstructured textual corpora. We either extract some collection-based statistics from the sources in the set $\mathcal{S}$ (for computing the importance features associated with the explicit query concepts in the set $\mathcal{X}_Q$ — see Section 2.2.1), or use them to perform pseudo-relevance feedback (for computing the expansion scores associated with the expansion terms in the set $\mathcal{E}_Q^{init}$ — see Section 2.2.2).

It is theoretically possible to use the same information sources for deriving both the importance features and the expansion scores. In practice, however, a single external source is commonly better suited for only one of these tasks. For instance, the Google N-grams source (a large collection of web n-gram counts) is useful for concept weighting, but not for query expansion. On the other hand, an entire external document collection such as Wikipedia is more suitable for query expansion.

Accordingly, in Table 2 we provide a list of external information sources along with their usage. For sources used for concept weighting, Table 2 defines the extracted importance features, which are similar to features used in previous work [4, 6, 7, 20, 43]. For sources used for query expansion, Table 2 defines a unit of retrieval, which is used for pseudo-relevance feedback from the source. As external sources for query expansion, we use, in addition to the retrieval corpus, the heading text and the anchor text extracted from ClueWeb09, a large, publicly available web collection[4], as well as an English Wikipedia corpus.

As an example of the role that the external sources may play in query formulation, Table 3 demonstrates the expansion terms derived from the external information sources for the query *"toxic chemical weapon"*. Note that the *Combined* column in Table 3, which is the output of the process described in Section 2.2, includes expansion terms which are more relevant and address more of the query aspects than those produced by any individual source. For instance, it includes the terms *russia*, *agent*, *mustard* and *warfare*, which do not appear in the top terms obtained via pseudo-relevance feedback on the retrieval corpus. As a result, in this case, our query formulation approach improves the retrieval effectiveness by 33% over a method that uses pseudo-relevance feedback with the retrieval corpus, and by 14% over a method that uses pseudo-relevance feedback with Wikipedia.

[4] `http://boston.lti.cs.cmu.edu/clueweb09/`

4. RELATED WORK

The work presented in this paper integrates insights from several research areas into a unified and principled query formulation framework. In this section, we provide a brief overview of these research areas, and their relation to our work.

First, our framework is based on concept matching, rather than simple term matching used in the standard bag-of-words retrieval models [33, 35]. As such, it draws on the current research that goes beyond terms and utilizes phrases, proximities and term spans for information retrieval [28, 24, 31, 39]. Specifically, we adopt the *sequential dependence model* first proposed by Metzler and Croft [28], which incorporates single terms, adjacent bigram phrases and proximities.

Second, our framework is inspired by the recent research that leverages external information sources for supervised term and concept weighting [4, 6, 7, 20, 18, 14, 27, 42]. This research is mainly motivated by the need to address the challenge of retrieval with verbose queries, which often mix important and redundant concepts [3, 4, 20, 42]. The novelty of our work compared to this previous research is that, in addition to concept weighting, we use the external information sources for selecting helpful and diverse expansion terms.

Finally, our method uses pseudo-relevance feedback for query expansion, a practice which has a long and successful history in information retrieval (e.g., [8, 17, 25, 29, 44] to name just a few). Most of this research, however, has not been applied on the scale of web corpora, and only uses the retrieval corpora as a source for pseudo-relevance feedback. More recently, researchers started to examine the benefits

⟨*title*⟩	*Robust04*		*Gov2*		*ClueWeb-B*	
	nDCG@20	MAP	nDCG@20	MAP	nDCG@20	MAP
SD	41.78	25.85	43.24	30.90	21.36	19.37
WSD	42.02	26.10	44.06	31.68	22.20	20.23
MSF[10]	$\mathbf{44.13}^{w}_{s}$	$\mathbf{30.49}^{w}_{s}$ *(+17.9/+16.9)*	**44.91**	$\mathbf{34.35}^{w}_{s}$ *(+11.2/+8.4)*	$\mathbf{25.76}^{w}_{s}$	$\mathbf{23.96}^{w}_{s}$ *(+23.7/+18.4)*

⟨*desc*⟩	*Robust04*		*Gov2*		*ClueWeb-B*	
	nDCG@20	MAP	nDCG@20	MAP	nDCG@20	MAP
SD	40.85	25.70	40.69	27.23	17.47	12.90
WSD	43.16	27.82	41.74	28.53	18.78	13.81
MSF[10]	$\mathbf{44.86}^{w}_{s}$	$\mathbf{30.68}^{w}_{s}$ *(+19.4/+10.3)*	$\mathbf{43.13}_{s}$	$\mathbf{31.10}^{w}_{s}$ *(+14.2/+9.0)*	$\mathbf{20.32}_{s}$	$\mathbf{15.23}^{w}_{s}$ *(+18.0/+10.3)*

Table 4: Comparison with the query weighting methods. Statistically significant difference of MSF over the baselines are marked using s and w, for SD and WSD baselines, respectively. Best result per column is marked by boldface. The numbers in parenthesis indicate improvement over SD and WSD baselines, respectively.

of external information sources for pseudo-relevance feedback. Diverse sources such as Wikipedia [5, 26, 22, 45], large web and news corpora [12] and social bookmarking data [23] were found to be beneficial for document retrieval on both newswire and web corpora.

While each of the above parts of query formulation process (namely, concept detection, concept weighting and query expansion) has been extensively studied, there is still a lack of holistic approaches to query formulation framework, which integrate the above areas of research. Developing such an approach is our goal in this paper.

5. EVALUATION

In this section, we describe the details of our experimental evaluation. First, in Section 5.1 we explain the experimental setup used for our experiments. In Section 5.2 and Section 5.3 we compare the proposed query formulation framework to several baselines that use query weighting and/or query expansion. In Section 5.4 and Section 5.5 we further analyze our method, and explore its sensitivity to the number of expansion terms and its robustness. Finally, in Section 5.6 we examine the impact of query formulation with multiple information sources on the diversity of the retrieved results.

5.1 Experimental Setup

The retrieval experiments described in this section are implemented using Indri, an open-source search engine [41]. The structured query language implemented by Indri natively supports multiple concept types, including exact phrases and proximity matches, as well as custom term weighting schemes. As a result, Indri provides a flexible and convenient platform for evaluating the performance of the proposed query formulation method.

Table 5 presents a summary of the TREC corpora[5] used in our experiments. The corpora vary both by type (*Robust04* is a newswire collection, *Gov2* is a crawl of the .gov domain, and *ClueWeb-B* is a set of pages with the highest crawl priority derived from a large web corpus), number of documents, and number of available topics, thereby providing a diverse experimental setup for assessing the robustness of our query formulation method.

During indexing and retrieval, both documents and queries are stemmed using the Krovetz stemmer, which is a "light" stemmer, as it makes use of inflectional linguistic morphology [15]. The Krovetz stemmer is especially suitable for web collections (e.g., *ClueWeb-B*) where aggressive stemming can decrease precision at top ranks [32]. Stopword removal is performed on both documents and queries using the standard INQUERY stopword list. The free parameter μ in the concept matching function $f(\kappa,D)$ (see Equation 2) is set to 2500, according to the default Indri configuration of the Dirichlet smoothing parameter.

[5] `http://trec.nist.gov/`

Name	# Docs	Topic Numbers
Robust04	528,155	301-450, 601-700
Gov2	25,205,179	701-850
ClueWeb-B	50,220,423	1-100

Table 5: Summary of TREC collections and topics used for evaluation.

⟨*title*⟩	discovery channel store
⟨*desc*⟩	Find locations and information about Discovery Channel stores and types of products they sell.

Figure 1: An example of ⟨*title*⟩ and ⟨*desc*⟩ portions of a TREC topic.

The performance of our query formulation method is compared to a number of state-of-the-art retrieval methods that perform concept weighting, query expansion or both. In all the comparisons, our query formulation method is denoted `MSF[N]` (*Multiple Source Formulation*), where N is the number of terms used for query expansion. As comparison metrics, we use both the mean average precision (*MAP*) of the entire ranked list, and the normalized discounted cumulative gain at the top ranks (*nDCG@20*).

The optimization of the free parameters for both the proposed query formulation method and all the baseline methods is done using 3-fold cross-validation with mean average precision (*MAP*) as the target metric. The statistical significance of differences in the performance of the `MSF` method with respect to other baselines is determined using a two-sided Fisher's randomization test [38] with 25,000 permutations and $\alpha < 0.05$.

As was shown in previous work [3, 4, 6, 7, 18], the impact of query formulation techniques varies significantly across queries of different length. In general, more verbose queries are expected to benefit more from effective weighting of both explicit query concepts and expansion terms, since they are more likely to contain concepts of varying importance.

⟨*title*⟩	*Robust04*		*Gov2*		*ClueWeb-B*	
	nDCG@20	MAP	nDCG@20	MAP	nDCG@20	MAP
LCE[10]	43.77	28.89	43.26	32.59	21.95	20.90
LCE-WP[10]	**44.63**	28.93	43.98	31.90	25.43	23.47
PQE[10]	44.23	29.06	44.58	33.64	21.94	20.82
MSF[10]	44.13	$\mathbf{30.49}^{p}_{l,lw}$*(+4.9)*	$\mathbf{44.91}_{l}$	$\mathbf{34.35}^{p}_{l,lw}$*(+2.1)*	$\mathbf{25.76}^{p}_{l}$	$\mathbf{23.96}^{p}_{l}$*(+2.1)*

⟨*desc*⟩	*Robust04*		*Gov2*		*ClueWeb-B*	
	nDCG@20	MAP	nDCG@20	MAP	nDCG@20	MAP
LCE[10]	42.24	28.05	41.10	30.14	18.17	14.00
LCE-WP[10]	44,38	29.08	41.45	28.70	19.90	14.52
PQE[10]	44.32	29.56	42.86	30.96	18.35	14.10
MSF[10]	$\mathbf{44.86}_{l}$	$\mathbf{30.68}^{p}_{l,lw}$*(+3.8)*	$\mathbf{43.13}_{l,lw}$	$\mathbf{31.10}_{lw}$*(+0.5)*	$\mathbf{20.32}^{p}_{l}$	$\mathbf{15.23}^{p}_{l,lw}$*(+4.8)*

Table 6: Comparison with the query expansion methods. Statistically significant difference of MSF over the baselines are marked using l, lw, and p, for LCE, LCE-WP and PQE baselines, respectively. Best result per column is marked by boldface. The numbers in parenthesis indicate improvement over the baseline with the best performance.

Thus, to test the performance of MSF across different query types we treat the ⟨*title*⟩ and the ⟨*desc*⟩ portions of TREC topics as two separate query sets in our experiments. The ⟨*title*⟩ and the ⟨*desc*⟩ versions of each query represent the same information need, but differ in their level of verbosity. The ⟨*title*⟩ query is a short keyword query, while the ⟨*desc*⟩ query is a verbose natural language description of the information need. Figure 1 shows an example of ⟨*title*⟩ and ⟨*desc*⟩ queries for a standard TREC topic.

5.2 Comparison to Query Weighting Methods

In this section, we compare the retrieval effectiveness of our query formulation method, MSF, which performs both concept weighting and query expansion using external information sources (see Section 3 for details of the sources used), to the performance of the methods that perform query weighting alone.

Our first baseline is the sequential dependence model, which was first proposed by Metzler and Croft [28]. The sequential dependence model, denoted SD, uses the same explicit query concepts (query terms, and adjacent bigram phrases and proximities) as our query formulation method. SD assigns fixed weights $\lambda(\kappa, Q)$ to all the concepts of the same type.

Following Metzler and Croft [28] we set these weights to 0.8, 0.1 and 0.1 for query terms, phrases and proximities, respectively. This parameter setting has been found to lead to a performance that is significantly superior to that of the standard bag-of-words models such as query likelihood or BM25 [28] for both TREC collections [28, 29] and web corpora [6].

Our second baseline is a weighted variant of the sequential dependence model, first proposed by Bendersky et al. [6]. The weighted sequential dependence model, denoted WSD, uses a combination of external sources for query concept weighting, but does not perform query expansion. It has been shown to attain significant gains over the SD method, especially for verbose queries, which contain concepts of varying importance [6, 7]. WSD uses the method described in Section 2.2.1 for estimating explicit concept weights, and the same set of sources for query weighting as in Table 2. As such, WSD allows us to examine the benefit provided by the *expansion* stage of the query formulation.

Table 4 compares the performance of the above baselines (SD and WSD) and our query formulation method, MSF, with 10 expansion terms. Table 4 unequivocally demonstrates the importance of query expansion for query formulation.

MSF is always more effective than the baselines that do not perform query expansion, and in all cases its improvements are statistically significant. These improvements are consistent across retrieval metrics, corpora and query types. The largest improvements are observed for short web queries (*ClueWeb-B*, ⟨*title*⟩ queries) where our method achieves effectiveness gains over 17% at high ranks (*nDCG@20* metric) and over 18% improvement in the quality of the entire ranked list (*MAP* metric).

5.3 Comparison to Query Expansion Methods

After comparing the effectiveness of the MSF method against methods that do not perform query expansion, in this section we focus on comparing its performance to that of current state-of-the-art query expansion methods.

First, we make use of the Latent Concept Expansion (LCE) method, which was shown to be a state-of-the query expansion method that uses a single collection [29, 16, 19]. Similarly to MSF, LCE uses Equation 5 to select expansion terms from the collection on which the pseudo-relevance feedback is performed.

As baselines, we implement two variants of LCE. The first baseline is denoted LCE. It is the standard version of Latent Concept Expansion, which performs the pseudo-relevance feedback on the retrieval corpus.

The second baseline is denoted LCE-WP. LCE-WP performs the pseudo-relevance feedback on Wikipedia, rather than the retrieval corpus. LCE-WP is based on some recent work that shows that query expansion using Wikipedia corpus can be beneficial, especially for short ambiguous queries over large web collections [5, 22, 26, 45].

In addition to the LCE-based baselines, we use the Parameterized Query Expansion method [7] as a baseline. This method, denoted PQE, combines explicit concept weighting and expansion term weighting in a unified framework that uses external information sources. The main difference between the PQE and the MSF methods, is that the former uses the external sources solely for weighting purposes, while the latter uses them also for expansion term selection.

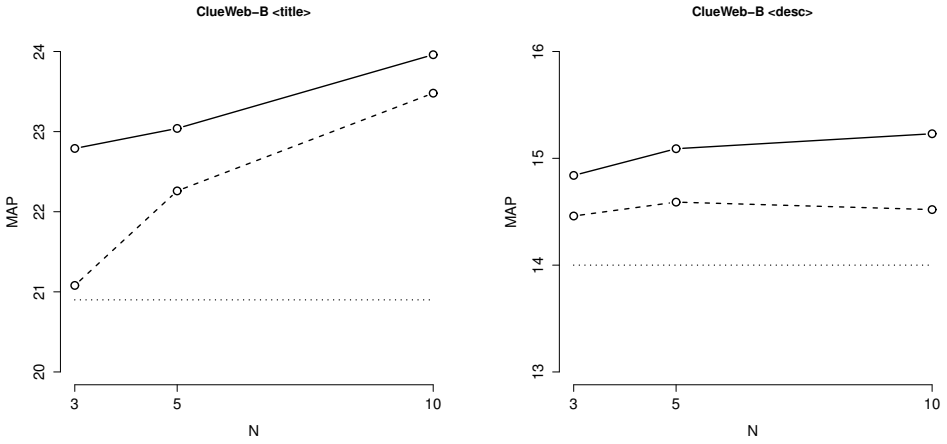

Figure 2: Varying the number of expansion terms (*ClueWeb-B* corpus). Dotted line indicates the performance of LCE[10]. Dashed and solid lines represent the performance of LCE-WP[N] and MSF[N], respectively.

Table 6 compares the effectiveness of the three baselines described above (LCE, LCE-WP and PQE) to the proposed MSF method, when 10 expansion terms are used. This comparison highlights the different positive aspects of MSF method.

The main observation from the Table 6 is that MSF is almost always more effective than any of the three baselines (except for *nDCG@20* for *Robust04*, where it is statistically indistinguishable from other methods). In contrast to the baselines, the performance of MSF is stable across corpora and query types. In comparison, the performance of the baselines is not as consistent. For instance, LCE-WP is more effective than LCE for *Robust04* and *ClueWeb-B*, but less effective for *Gov2*. Similarly, PQE outperforms LCE-based baselines for *Robust04* and *Gov2* corpora, but is not as effective for the *ClueWeb-B* corpus.

In addition, Table 6 clearly demonstrates the importance of using the external information sources for both concept weighting and expansion term selection. Compared to PQE, which uses the external sources solely for weighting purposes, MSF achieves significantly better performance on all metrics. This is especially evident in the case of the *ClueWeb-B* corpus, for which expansion using the retrieval corpus attains only marginal gains. For the *ClueWeb-B* corpus, PQE achieves merely a 2% gain over the WSD baseline for $\langle title \rangle$ queries, while MSF achieves over 18% gain (see Table 4). It is clear that in this case, using multiple sources for selecting the expansion terms, in addition to concept weighting, is highly beneficial.

Finally, Table 6 shows that the synergy of concept weighting and expansion term selection using external sources as performed by the MSF is superior to the ad-hoc approach that simply uses an external corpus (e.g., Wikipedia) for query expansion. MSF is more stable than LCE-WP across all collections, and is more effective even for the *ClueWeb-B* corpus, where expansion with Wikipedia was shown to be a highly effective strategy [5, 26].

5.4 Number of Expansion Terms

Massive query expansion with tens or even hundreds of terms, as is often done in TREC evaluation [8, 12] is not suitable for the scenario of web search, where the size of the retrieval corpus is large, and users expect low query latencies. Accordingly, in this section we explore the effect of query expansion with very few expansion terms, to demonstrate the scalability of the MSF method for web corpora.

In Figure 2 we plot the effectiveness (in terms of *MAP*) of query formulation methods that have the best performance for the *ClueWeb-B* corpus – LCE-WP and MSF– when using the 3, 5 and 10 highest weighted expansion terms. For comparison, we also plot the effectiveness of a standard query expansion method, LCE with 10 terms.

First, Figure 2 clearly demonstrates the superiority of both LCE-WP and MSF compared to LCE, even with fewer expansion terms. We can also see from Figure 2 that the superiority of the proposed MSF method over the LCE-WP method, which uses Wikipedia for query expansion, is not limited to the scenario in Table 6, where 10 expansion terms are used. The effectiveness gains of MSF over LCE-WP are consistent with minimal query expansion (3 or 5 additional terms) as well. For instance, when only 3 terms are used for query expansion, MSF achieves around 8% and 3% improvement over LCE-WP for $\langle title \rangle$ and $\langle desc \rangle$ queries, respectively.

Overall, the results in Figure 2 showcase the ability of the MSF method to produce both effective and compact query formulations, which could potentially scale to real world web search scenarios.

5.5 Robustness

In Table 6 we have shown that the MSF method significantly improves the overall performance compared to Latent Concept Expansion with the retrieval corpus (LCE) and Wikipedia (LCE-WP), and Parameterized Query Expansion (PQE). As discussed in Section 5.3, LCE-WP is a highly effective method for the *ClueWeb-B* corpus, while LCE and PQE are more suitable for the other two corpora. In this section, we analyze the *robustness* of MSF, compared to these methods. Following previous work [29], we define the robustness of the method as the number of queries improved or hurt (and by how much – in terms of *MAP*) as the result of the application of the method. A highly robust expansion technique will significantly improve many queries and only minimally hurt a few.

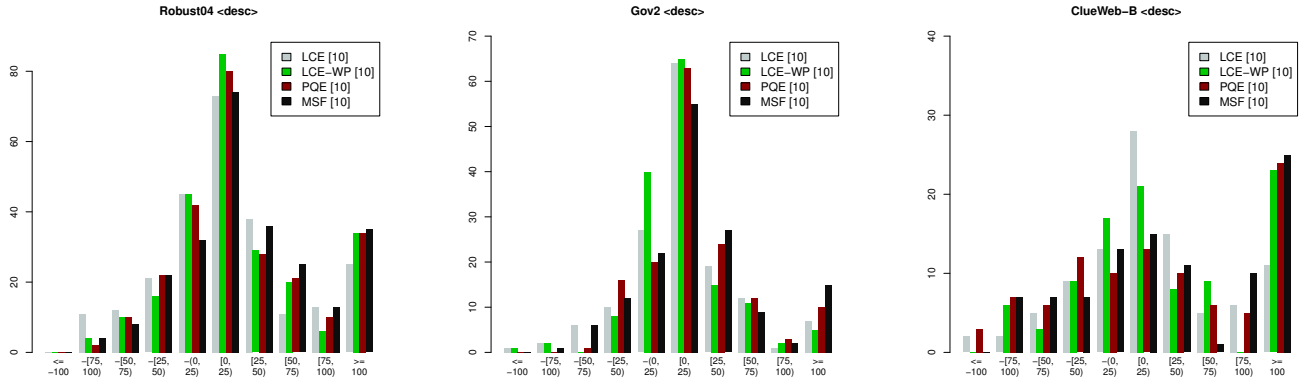

Figure 3: Robustness of the LCE, LCE-WP and MSF methods for the $\langle desc \rangle$ queries w.r.t. the SD method.

$\langle title \rangle$	α-nDCG@20	S-Recall@20	MAP-IA
WSD	22.36	46.04	9.20
PQE[10]	21.07	42.47	9.39
LCE-WP[10]	24.51	46.91	11.00
MSF[10]	$\mathbf{25.85}^{lw}_{w,p}$	$\mathbf{48.94}^{lw}_{p}$	$\mathbf{11.25}_{w,p}$

Table 7: Result diversification performance (*ClueWeb-B*). Statistically significant difference of MSF over the baselines are marked using w, p, and lw, for WSD, PQE and LCE-WP baselines, respectively. Best result per column is marked by boldface.

Figure 3 provides an analysis of the robustness of LCE, LCE-WP, PQE and MSF for the $\langle desc \rangle$ queries (which are, in general, harder than the $\langle title \rangle$ queries and benefit more from query expansion) with 10 expansion terms. The histograms in Figure 3 show, for various ranges of relative decreases or increases in *MAP*, the number of queries that were hurt or improved with respect to the SD baseline (see Table 4).

Figure 3 demonstrates that MSF is more robust compared to the other three methods. In all cases, MSF improves the performance of more queries than both LCE, LCE-WP and PQE. For instance, for the *Robust04* collection, MSF improves the performance of 72% of the queries w.r.t. SD, compared to 62%, 67% and 68% of the queries improved by LCE, LCE-WP and PQE respectively. Similar improvements are observed for the other two collections.

In addition, the MSF method is, on average, less likely to decrease the query performance, compared to the other methods. It hurts less queries than LCE and LCE-WP on all collections, and less queries than PQE on all collections except *Gov2*, where both methods hurt around 27% of the queries.

5.6 Impact on Result Diversification

Recently, result diversification in web search has become an active research topic [1, 9, 10, 36, 37]. Since web search queries are often underspecified and/or ambiguous, diversifying the search results may assist users with varying intents in finding relevant information in a single ranked list returned by the search engine. Due to the research interest in this problem, result diversification was chosen as a search task during the 2009 and 2010 TREC Web Tracks [10].

Effective result diversification is often achieved by *inter-query* approaches. These approaches combine results from queries that are found to be related to the original user query (e.g., through access to the query suggestions proposed by commercial search engines [36, 37]). However, even in the inter-query approaches, the retrieval effectiveness and diversity performance of each single query is important for obtaining the optimal diversification results [37].

Therefore, in this section we examine *intra-query* result diversification, i.e., the diversity performance that can be achieved by using the original user query alone. To this end, we compare the performance of the three best-performing baselines from Table 4 and Table 6 (WSD, PQE and LCE-WP) to that of the MSF method in terms of three standard diversity metrics. These diversity metrics include metrics that examine the diversity at the top ranks (α-nDCG and subtopic recall at rank 20) [9, 10], as well as a metric that measures the diversity of the entire ranked list (intent-aware MAP) [1].

Table 7 demonstrates the comparison of the result diversification performance of the different methods on the $\langle title \rangle$ queries for the *ClueWeb-B* collection[6]. Overall, MSF achieves the best diversity performance, especially for the diversity at the top ranks, where it achieves over 6% improvement over LCE-WP, the best-performing baseline.

In the context of search result diversification, it is interesting to note that previous work suggested that query expansion with the retrieval corpus may reduce diversity at top ranks [9]. The comparison between the WSD and the PQE baselines in Table 7 is in line with this finding. In contrast to the expansion with the retrieval corpus alone, the proposed MSF method helps to improve the diversity of the search results, since it combines expansion terms from different sources.

6. CONCLUSIONS

In this paper, we introduced a novel framework for query formulation. This framework synthesizes, in a principled and effective manner, arbitrary concept matches, concept weighting and query expansion. Our query formulation ap-

[6]We do not include the $\langle desc \rangle$ queries in our diversification performance analysis, since these are verbose and non-ambiguous queries that fully specify the user intent.

proach leverages external sources of information such as web n-gram counts, anchor and heading text extracted from a large web corpus, and articles and titles from Wikipedia for weighting the explicit query concepts as well as selecting relevant and diverse set of weighted expansion terms.

We perform a thorough empirical evaluation of our query formulation approach, `MSF`. Our experimental results unequivocally demonstrate the superiority of `MSF` to several state-of-the-art baselines that perform concept weighting, query expansion or both. Further analysis of the performance of the `MSF` method shows that it is highly robust across corpora and query types, enables compact query representations by reducing the number of required expansion terms, and improves the diversity of the retrieved results.

7. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by ARRA NSF IIS-9014442. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *Proc. of WSDM*, pages 5–14, 2009.

[2] G. Amati and C. J. Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4):357–389, October 2002.

[3] N. Balasubramanian, G. Kumaran, and V. R. Carvalho. Exploring reductions for long web queries. In *Proc. of SIGIR*, pages 571–578, 2010.

[4] M. Bendersky and W. B. Croft. Discovering key concepts in verbose queries. In *Proc. of SIGIR*, pages 491–498, 2008.

[5] M. Bendersky, D. Fisher, and W. B. Croft. UMass at TREC 2010 Web Track: Term Dependence, Spam Filtering and Quality Bias. In *Proc. of TREC-10*, 2011.

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

[7] M. Bendersky, D. Metzler, and W. B. Croft. Parameterized concept weighting in verbose queries. In *Proc. of SIGIR*, 2011.

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

[9] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proc. of SIGIR*, pages 659–666, 2008.

[10] C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 Web Track. In *Proc. of TREC-09*, 2010.

[11] W. B. Croft, M. Bendersky, H. Li, and G. Xu. Query representation and understanding workshop report. SIGIR Forum, December 2010.

[12] F. Diaz and D. Metzler. Improving the estimation of relevance models using large external corpora. In *Proc. of SIGIR*, pages 154–161, 2006.

[13] J. Guo, G. Xu, H. Li, and X. Cheng. A unified and discriminative model for query refinement. In *Proc. of SIGIR*, pages 379–386, New York, NY, USA, 2008.

[14] D. Hiemstra. Term-specific smoothing for the language modeling approach to information retrieval: the importance of a query term. In *Proc. of SIGIR*, pages 35–41, 2002.

[15] R. Krovetz. Viewing morphology as an inference process. In *Proc. of SIGIR*, pages 191–202, 1993.

[16] H. Lang, D. Metzler, B. Wang, and J.-T. Li. Improved latent concept expansion using hierarchical markov random fields. In *Proc. of CIKM*, pages 249–258, 2010.

[17] V. Lavrenko and W. B. Croft. Relevance Models in Information Retrieval. In W. B. Croft and J. Lafferty, editors, *Language modeling for Information Retrieval*, pages 11–56. Kluwer, 2003.

[18] M. Lease. An improved markov random field model for supporting verbose queries. In *Proc. of SIGIR*, pages 476–483, 2009.

[19] M. Lease. Incorporating relevance and pseudo-relevance feedback in the markov random field model. In *Proc. of TREC-08*, 2009.

[20] M. Lease, J. Allan, and W. B. Croft. Regression rank: Learning to meet the opportunity of descriptive queries. In *Proc. of ECIR*, pages 90–101, 2009.

[21] H. Li. *Learning to Rank for Information Retrieval and Natural Language Processing*. Morgan and Claypool Publishers, 2011.

[22] Y. Li, W. P. R. Luk, K. S. E. Ho, and F. L. K. Chung. Improving weak ad-hoc queries using wikipedia as external corpus. In *Proc. of SIGIR*, pages 797–798, 2007.

[23] Y. Lin, H. Lin, S. Jin, and Z. Ye. Social annotation in query expansion: a machine learning approach. In *Proc. of SIGIR*, pages 405–414, 2011.

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

[25] Y. Lv and C. Zhai. Positional relevance model for pseudo-relevance feedback. In *Proc. of SIGIR*, pages 579–586, 2010.

[26] R. McCreadie, C. Macdonald, I. Ounis, J. Peng, and R. L. T. Santos. University of Glasgow at TREC 2009: Experiments with Terrier. In *Proc. of TREC-09*, 2010.

[27] Q. Mei, H. Fang, and C. Zhai. A study of Poisson query generation model for information retrieval. In *Proc. of SIGIR*, pages 319–326, 2007.

[28] D. Metzler and W. B. Croft. A Markov random field model for term dependencies. *Proc. of SIGIR*, pages 472–479, 2005.

[29] D. Metzler and W. B. Croft. Latent concept expansion using markov random fields. *Proc. of SIGIR*, pages 311–318, 2007.

[30] D. Metzler and W. B. Croft. Linear feature-based models for information retrieval. *Information Retrieval*, 10(3):257–274, 2007.

[31] G. Mishne and M. de Rijke. Boosting Web Retrieval Through Query Operations. In *Proc. of ECIR*, pages 502–516, 2005.

[32] F. Peng, N. Ahmed, X. Li, and Y. Lu. Context sensitive stemming for web search. In *Proc. of SIGIR*, pages 639–646, 2007.

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

[34] S. Robertson, H. Zaragoza, and M. Taylor. Simple bm25 extension to multiple weighted fields. In *Proc. of CIKM*, pages 42–49, 2004.

[35] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proc. of SIGIR*, pages 232–241, 1994.

[36] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *Proc. of WWW*, pages 881–890, 2010.

[37] R. L. Santos, C. Macdonald, and I. Ounis. Intent-aware search result diversification. In *Proc. of SIGIR*, pages 595–604, 2011.

[38] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proc. of CIKM*, pages 623–632, 2007.

[39] R. Song, M. Taylor, J.-R. Wen, H.-W. Hon, and Y. Yu. Viewing term proximity from a different perspective. In *Proc. of ECIR*, pages 346–357, 2008.

[40] K. Sparck Jones. *Document retrieval systems*, chapter A statistical interpretation of term specificity and its application in retrieval, pages 132–142. Taylor Graham Publishing, 1988.

[41] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language model-based search engine for complex queries. In *Proc. of IA*, 2004.

[42] K. M. Svore, P. H. Kanani, and N. Khan. How good is a span of terms?: exploiting proximity to improve web retrieval. In *Proc. of SIGIR*, pages 154–161, 2010.

[43] L. Wang, D. Metzler, and J. Lin. Ranking under temporal constraints. In *Proc. of CIKM*, pages 79–88, 2010.

[44] J. Xu and W. B. Croft. Query expansion using local and global document analysis. *Proc. of SIGIR*, pages 4–11, 1996.

[45] Y. Xu, G. J. F. Jones, and B. Wang. Query dependent pseudo-relevance feedback based on wikipedia. In *Proc. of SIGIR*, pages 59–66, 2009.

[46] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.