Information Processing and Management 54 (2018) 219–239

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Verbosity normalized pseudo-relevance feedback in information retrieval

Seung-Hoon Na[1,a], Kangil Kim[*,b]

[a] Dept. of Computer Science, Chonbuk National University, South Korea
[b] Dept. of Computer Science, Konkuk University, South Korea

ARTICLE INFO

Keywords:
Pseudo-relevance feedback
Verbosity normalization
Scope normalization
Term frequency

ABSTRACT

Document length normalization is one of the fundamental components in a retrieval model because term frequencies can readily be increased in long documents. The key hypotheses in literature regarding document length normalization are the verbosity and scope hypotheses, which imply that document length normalization should consider the distinguishing effects of verbosity and scope on term frequencies. In this article, we extend these hypotheses in a pseudo-relevance feedback setting by assuming the *verbosity hypothesis on the feedback query model*, which states that the verbosity of an expanded query should not be high. Furthermore, we postulate the following two effects of document verbosity on a feedback query model that easily and typically holds in modern pseudo-relevance feedback methods: 1) the *verbosity-preserving effect*: the query verbosity of a feedback query model is determined by feedback document verbosities; 2) the *verbosity-sensitive effect*: highly verbose documents more significantly and unfairly affect the resulting query model than normal documents do. By considering these effects, we propose verbosity normalized pseudo-relevance feedback, which is straightforwardly obtained by replacing original term frequencies with their verbosity-normalized term frequencies in the pseudo-relevance feedback method. The results of the experiments performed on three standard TREC collections show that the proposed verbosity normalized pseudo-relevance feedback consistently provides statistically significant improvements over conventional methods, under the settings of the *relevance model* and *latent concept expansion*.

1. Introduction

Term frequency is one of the most important components in document ranking. However, term frequencies are readily increased when documents are long and there are various document lengths. Therefore, when a naive method is implemented using term frequencies, the resulting retrieval model typically prefers long documents to short documents. Document length normalization has been investigated extensively in literature to prevent unfair preference for long documents.

The key hypotheses in document length normalization are the verbosity and scope hypotheses. These state that document length is affected by two factors, *verbosity* and *scope*, as noted in (Robertson & Walker, 1994; Robertson & Zaragoza, 2009):

1) **Verbosity hypothesis**: "*Some authors are simply more verbose, using more words to say the same thing*" (Robertson & Zaragoza, 2009).

2) **Scope hypothesis**: "*Some authors have more to say: they may write a single document containing or covering more*

* Corresponding author.
E-mail addresses: nash@jbnu.ac.kr (S.-H. Na), kikim01@konkuk.ac.kr (K. Kim).
[1] Tel: (82) 031-270-1822

https://doi.org/10.1016/j.ipm.2017.09.006
Received 4 April 2016; Received in revised form 26 September 2017; Accepted 29 September 2017
Available online 23 November 2017

ground" (Robertson & Zaragoza, 2009).

In other words, verbosity helps to significantly increase term frequencies, while scope mostly involves the creation of a new word, rather than boosting term frequencies. The distinguishing effects of verbosity and scope on term frequencies guide us to design a document normalization function; the verbosity hypothesis suggests that term frequency should be strongly penalized by document length, i.e., "full penalization", whereas the scope hypothesis suggests the opposite, i.e., "non-penalization", as in (Robertson & Zaragoza, 2009).

In this article, we discuss the verbosity and scope hypotheses on the pseudo-relevance feedback (PRF) setting by considering the effect of document length and verbosity on query expansion and reweighting. In our discussion, without loss of generality, expanded queries resulting from PRF are referred to as *feedback query models*, or simply, "query models," following the terminology for the KL-divergence framework (Lafferty & Zhai, 2001)[2].

Our starting assumption is the *query verbosity hypothesis for a feedback query model*, which states that a feedback query model must be as compact as possible. This is equivalent to stating that its verbosity should not be high, when conveying the same relevant topics. More specifically:

• **Verbosity hypothesis of a feedback query model**: When it is not ensured that a retrieval model can effectively handle verbose queries in comparison to compact queries, the verbosity of a feedback query model should not be high. Formally, let $v(q)$ be the verbosity of feedback query model q[3] and V_R be the set of *relevant* terms to an original query that is obtained from a set of relevant documents. Suppose that q_1 and q_2 are feedback query models estimated from a subset of *relevant* documents[4] and all terms of q_1 and q_2 are relevant. The verbosity hypothesis of a feedback query model is presented as:

Given $q_1 \in V_R{}^*$ *and* $q_2 \in V_R{}^*$, *if* $v(q_1) \leq v(q_2)$, q_1*should be preferred over* q_2*as a feedback query model for second-stage retrieval*[5].

An argument can be made to justify the verbosity hypothesis on the query model. Modern retrieval models are likely more effective on keyword queries than verbose queries, when the number of relevant terms is similar. This is because verbose queries might likely contain less topically relevant terms or common terms, thereby negatively affecting the retrieval performance for second-stage retrieval. This argument is partly similar to the inverse document frequency (IDF) effect in (Clinchant & Gaussier, 2013), which is one of the important constraints that need to hold in a PRF method, as extensively explored in (Hazimeh & Zhai, 2015).

As a next step, we postulate two effects issuing from document verbosity on feedback query models that readily and commonly hold in modern PRF methods: namely, the verbosity-preserving effect and the verbosity-sensitive effect.

• **Verbosity-preserving effect on a feedback query model**: The *verbosity-preserving effect* refers to the tendency of the verbosity of a feedback query model to be proportional to the verbosities in feedback documents. In other words, the verbosity of feedback documents dominantly determines the query verbosity of a feedback query model. A PRF method that has the verbosity-preserving effect on a feedback query model is referred to as a *verbosity-preserving PRF* method. In a verbosity-preserving PRF method, because the verbosity of a feedback query model is not free from the verbosity of feedback documents, it is important to use compact documents (i.e., feedback documents with low verbosity) to derive a compact query model.

• **Verbosity-sensitive effect on a feedback query model**: The *verbosity-sensitive effect* refers to the fact that the weight of the expansion term is more dominantly affected by verbose documents than other documents in top-retrieved documents. A PRF method that has the verbosity-sensitive effect on a feedback query model is referred to as a *verbosity-sensitive PRF* method. When using a verbosity-sensitive PRF method, verbosity normalization needs to be applied separately to prevent a query model from relying heavily on verbose documents in a unfair manner, developing a *verbosity-robust* PRF method.

The verbosity-preserving effect guides us to employ a method for improving the retrieval effectiveness of second-stage retrieval by examining the verbosity of feedback documents. According to the verbosity hypothesis of a feedback query model, when using a verbosity-preserving PRF method, the less verbose the feedback documents, the more effective the second-stage retrieval is, when feedback documents are of similar levels of relevance. Therefore, a retrieval model that tends not to prefer verbose documents in top-ranked ones is advantageous over a method that likely prefers to verbose documents, unless the degree of the relevance of top-retrieved documents is not significantly different.

Next, the verbosity-sensitive effect encourages us to apply verbosity normalization to feedback document representations. This proposal is largely similar to the statement of the document length effect in (Clinchant & Gaussier, 2013).

Taking account of the aforementioned verbosity effects on a query model – i.e., the verbosity-preserving and verbosity-sensitive effects – this paper proposes the use of the existing two-stage normalization method in (Na, 2015) for PRF. This two-stage normalization method was originally proposed to elaborately model the different effects of verbosity and scope on term frequency by sequentially performing verbosity and scope normalization (Na, 2015). In verbosity normalization, term frequency is first pre-normalized by directly dividing according to the document's length, resulting in *verbosity-normalized document representation*. For scope normalization, these resulting normalized term frequency vectors are directly applied when defining the scoring function. The resulting scoring formula additionally uses the document scope as a document-specific value, which further generalizes an existing formula based on the documents length and leads to the formulation of *verbosity normalized* (**VN**) retrieval models.

The main method proposed in this paper is the application of generalized two-stage normalization to PRF – i.e., *verbosity*

[2] In this paper, a "feedback query model" refers to either a unigram language model or a probabilistic latent concept model, depending on whether the relevance model (Lavrenko & Croft, 2001) or latent concept expansion of (Metzler & Croft, 2007) is used as the PRF method.

[3] For the definition of verbosity, see Section 3.

[4] Note that we assume an ideal situation in our hypothesis, i.e., the feedback documents used to estimate the query model are relevant to a query (not pseudo-relevant). In addition, the feedback documents used for q_1 and q_2 are not necessarily the same.

[5] With the abuse of notation, we use "$q \in V_R{}^*$" to indicate that for any $w \in q$, $w \in V_R$.

normalized PRF (**VN-PRF**) – such that it can be extensively applicable to the relevance model (RM) of (Lavrenko & Croft, 2001) and latent concept expansion (LCE) of (Metzler & Croft, 2007). The applications to RM and LCE are straightforward, where original term frequencies in feedback documents or in all Markov random field (MRF) features are replaced with verbosity normalized ones, resulting in *verbosity normalized RM* (**VN-RM**) and *verbosity normalized LCE* (**VN-LCE**).

Experimental results show that the proposed VN-RM and VN-LCE offer statistically significant improvements over RM and LCE, respectively, achieving a 2–3% absolute increase in mean average precision (MAP). Specifically, the proposed VN-LCE achieved state-of-the-art performance with some test collections. These results suggest that PRF should be designed by addressing verbosity preservation and preference problems. Furthermore, we empirically and theoretically justify that our verbosity-normalized PRF methods can indeed handle the verbosity-preserving effect and relax the verbosity-sensitive effect, given the characteristics entailed from verbosity normalized retrieval models.

The remainder of this paper is organized as follows. Section 2 describes related works. Section 3 reviews the two-stage document length normalization method to obtain verbosity normalized retrieval models. Section 4 describes the detailed technique of deriving the proposed verbosity-normalized LCE. Section 5 presents the main results from the comparative analysis between existing PRF methods and the proposed VN-PRF methods under the verbosity-preserving and verbosity-sensitive effects. Sections 6 and 7 respectively present the experimental setting and results. Finally, Section 8 presents our conclusions.

2. Related work

2.1. Document length normalization

Document length normalization is a long-standing research area in information retrieval (Robertson & Walker, 1994; Robertson et al., 1995; Singhal et al., 1996; He & Ounis, 2003; Smucker & Allan, 2005; Na et al., 2008; Lv & Zhai, 2011a,b,c; Cummins & O'Riordan, 2012; Rousseau & Vazirgiannis, 2013a,b; Lipani et al., 2015; Na, 2015; Cummins et al., 2015; Cummins, 2016). In early works, Robertson and Walker (1994) introduced the verbosity hypothesis and the scope hypothesis and noted different effects of verbosity and scope on length normalization. They consequently developed a pivoted document length. It was first incorporated in the Okapi BM25 model (Robertson et al., 1995) and was further elaborated in the pivoted vector space model (Singhal et al., 1996). Using a pivoted length, a relaxed manner of penalization is enabled. Accordingly, the term frequency is not simply normalized by dividing it by the original document length but by dividing it by its pivoted length.

Similar types of normalizations have been used in language modeling approaches (Zhai & Lafferty, 2001a) and in divergence from randomness frameworks (Amati & Rijsbergen, 2002). This kind of relaxed normalization was formally generalized in (Fang et al., 2004, 2011), specifically in the axiomatic work of formally defining required constraints. For length normalization, Na et al. (2008) introduced verbosity-normalization constraints such that relevance scores are invariant after replicating document content k times. In Lv and Zhai (2011b), lower-bounding heuristics were proposed for term frequency normalization to avoid over-penalization of very long documents, motivated by the previous empirical observation of (Lv & Zhai, 2011a).

Regarding query length effects, Chung et al. (2006) incorporated a query length to pivoted normalization method. In addition, Cummins and O'Riordan (2012) introduced a query length normalization constraint to regularize the effect of the query length on scoring long documents, which has been further generalized by formulating a kind of scope normalization constraint when applying scope-broadening perturbation to a query (Cummins, 2016). A similar query length normalization constraint was formulated in (Lv, 2015). For query verbosity, Di Buccio et al. (2014) newly defined the task of automatically detecting verbose queries and proposed a query classification method based on decision tree using various query term features.

In Rousseau and Vazirgiannis (2013a), a set of various term frequency normalizations was comprised, including Okapi normalization and pivoted normalization, which showed improved performances. In (Rousseau & Vazirgiannis, 2013a), instead of using term frequencies, a graph-based model was proposed in which the term weight is the in-degree in the directed word graph of a document. Considering their graph formation setting, the term weight is determined by the number of unique terms in the (left-side) surrounding words of a given term. Furthermore, Lv and Zhai (2011a) proposed adaptive term frequency normalization by setting a retrieval parameter (k_1) in Okapi BM25 in a term-specific way. Lipani et al. (2015) introduced a term-specific estimation for a pivot parameter (b) of Okapi BM25 by incorporating average term frequencies.

The verbosity hypothesis and the scope hypothesis of (Robertson & Walker, 1994) suggest that a relevance score must be formulated by considering different aspects of verbosity and scope. In Na et al. (2008), verbosity and scope normalization heuristics were introduced to explicitly address the specific scoring problem resulting from the respective verbosity and scope hypotheses, although they focus on the ideal situation.

In addition, a multi-aspect TF (MATF) was proposed in (Paik, 2013) with consideration of different aspects of verbosity and scope on retrieval scores. It was then extended to a maximum value model (Paik, 2015). Another recent work (Na, 2015) generalized the previous work of (Na et al., 2008) by proposing two-stage document length normalization starting from a simple decomposition of document length to verbosity and scope, resulting in the VN retrieval models.

Independently to our work of (Na et al., 2008; Na, 2015), Cummins et al. (2015) proposed SPUD language models by explicitly modeling word burstiness based on Dirichlet compound process. Interestingly, one of the SPUD models is equivalent to the VN Dirichlet-prior smoothing of (Na, 2015). Despite of the strengths of their principled derivation of the SPUD model, including a novel method with using automatic estimation of a smoothing parameter, the concentration parameter in SPUD is fixed to the number of unique words given their minimum setting. It is not extended to another scope measure, such as perplexity used in (Kurland & Lee, 2005; Na et al., 2008; Bendersky et al., 2011a; Na, 2015). Given this equivalence, we refer to the SPUD model as a VN Dirichlet-

smoothing model using the number of unique words as the scope measure. More recently, Cummins (2016) focused on long queries and proposed a discriminative query model. It was shown that the SPUD model is effective for long queries, under discriminative query models with statistically significant improvement over all recently developed length-normalization methods, including lower-bounding normalization of (Lv & Zhai, 2011b) and MATF of (Paik, 2013).

2.2. Pseudo-relevance feedback

PRF has been shown to be effective in significantly improving the retrieval performance. In the language modeling approaches, the issue is how to estimate the *query language model* from the set of top-retrieved documents where two main approaches have been investigated for PRF: a *mixture model* (Zhai & Lafferty, 2001b) and RM (Lavrenko & Croft, 2001). In the mixture model, it is assumed that a word in the top-retrieved document is generated by the two component mixture models: a query model and a collection model. The query model is then estimated by maximizing the likelihood of the feedback documents. Second, in the RM, the query model is computed by the conditional probability of word w given a query q, formulated as $P(w|q)$, which is equal to $\Sigma_d P(w|d)P(d|q)$. Of the two methods, the relevance model is known to be more robust than the mixture model (Lv & Zhai, 2009b). The commonly used variant of the relevance model is RM3, which has been shown to be effective, as in the work of Lv and Zhai (2009a).

Lv and Zhai (2010) proposed a positional relevance model, which is an extension of RM, by assigning more importance to expansion terms that were closer to query words based on a positional language model (Lv & Zhai, 2009a). As a document typically consists of several different topics, positional relevance models are more likely to select relevant terms that are about a query topic, as compared to RM, by focusing less on irrelevant topics in feedback documents. In our view, Lv and Zhai (2010)'s work is a method of approximately applying scope normalization on PRF, whereas our work is a direct application of verbosity normalization to PRF[6].

Ye et al. (2010) explored Rocchio's method in the DFR framework and proposed a quality-biased PRF method by incorporating the quality scores of feedback documents with Rocchio's method, where more importance is assigned to a feedback document with higher quality. In Ye et al. (2010), the quality score of a feedback document was computed as its relevance score to the original query. Their quality-biased Rocchio's method deployed on the DFR framework exhibited statistically significant improvements over RM3 and the baseline PRF (of the DFR framework)[7]. Hui et al. (2011) further applied the quality-biased method to BM25 and reported similar improvements over RM3 and the baseline PRF.

Recently, Zhang et al. (2014) proposed bias-variance analysis methods as novel evaluation measures of query models. Karisani et al. (2016) proposed a novel form of term reweighting by exploiting the similarity of each top document to other top-retrieved documents.

Going beyond the term independence assumption, Metzler and Croft (2005) proposed MRF, which is the graphical model for matching queries and documents. The term dependency in a query is characterized as feature functions derived by applying retrieval models. MRF was further extended to a weighted sequential dependency model of (Bendersky et al., 2010) by using concept weighting methods, whereby a concept feature is further decomposed into concept important features. In (Bendersky et al., 2010), concept weightings use both collection-dependent and collection-independent features, which are obtained from various sources of collections, such as Google n-gram and Wikipedia.

LCE is the method of PRF under the setting of MRF. It is considered a generalization of RM3 by exploiting the original query dependency and making the query term weight discriminative (Metzler & Croft, 2007). Owing to its discriminative nature, the weights of expansion terms are not simply fixed by its (smoothed) generative probability and the relevance score of a document (the posterior probability of a document given a query). LCE was shown to be effective by making greater improvements over RM. LCE was further generalized to parameterized query expansion (PQE) of (Bendersky et al., 2011b), whereby concept weights are applied to both query concepts and latent concepts of expansion terms in PRF. As shown in (Bendersky et al., 2011a), PQE makes statistically significant improvements over LCE. An additional extension of LCE is LCE HMRF of (Lang et al., 2010), which uses the hierarchical relation between expansion terms and leads to significant improvements over LCE.

Our extension of LCE, named VN-LCE, is the resultant model resulting by from the applying application of two-stage normalization to LCE. VN-LCE is used to replace the top-retrieved documents retrieved by MRF with one resulting from the VN-MRF model. VN-LCE additionally uses term weights resulting from VN-MRF instead of ones from MRF. Despite this simplicity, the results are noticeable, showing significantly improved performances, by making it comparable to LCE_HMRF on some collections without the hierarchical relation.

Recently, Cummins et al. (2015) presented experimental results from examining the effects of PRF based on SPUD. However, the VN Markov random field framework with a perplexity-based scope measure has not been explored in (Cumminset al., 2015). Our works therefore differ from (Cumminset al., 2015)'s work in that we explore the effects of two-stage normalization under LCE and use the scope measure as perplexity (i.e., EntropyPower in (Na, 2015)), whereas Cummins et al. (2015) used RM3 with the number of unique terms as the scope measure.

[6] As verbosity is related to scope in our definition, scope normalization to PRF may be partly related to verbosity normalization to PRF.

[7] In the baseline PRF, first, a pseudo large feedback document is created from feedback documents; then, the weight of an expansion term is obtained by applying a DRF weighting scheme to the pseudo document.

3. Verbosity normalized retrieval models

In this section, we briefly review the two-stage normalization and the generalized technique for applying it to PRF.

3.1. Two-Stage normalization and VN models

The following are notations commonly used in this paper.

- V: $w_1, w_2, \cdots, w_{|V|}$, Set of all words
- N: Number of documents in a given collection
- C: A given collection, consisting of $d_1, \cdots, d_N$. Often, we also use C to refer to the concatenated representations of all documents in C.
- d (or q): A given document (or a query)
- $c(w, d)$ (or $c(w, q)$): Term frequency of word w in document d (or query q)
- $c(w, C)$: Term frequency of word w in collection C defined by $\Sigma_{d \in C} c(w, d)$
- $|d|$: Length of document d, defined by $\Sigma_{w \in V} c(w, d)$
- $|C|$: Length of collection C, defined by $\Sigma_{w \in V} c(w, C)$ (for brevity of notation, C is either the set of documents or the concatenated representation of documents, depending on context)
- $s(d)$: Scope of document d ($s(d) \leq |d|$)
- $v(d)$: Verbosity of document d
- *avgl, avgv, avgs*: Average length, verbosity, and scope, respectively, of documents in the collection.

Motivated by the verbosity-scope-hypotheses, we first assume that document length is decomposed into verbosity and scope as follows:

$$|d| = v(d)s(d) \tag{1}$$

As a result, $v(d)$ can be rewritten in terms of $s(d)$ and $|d|$ as follows:

$$v(d) = \frac{|d|}{s(d)} \tag{2}$$

The verbosity normalization is the process of making all documents have the same verbosity. In verbosity normalization, the original term frequency is normalized by dividing it by the verbosity of the document, resulting in *verbosity-normalized document representations* (i.e., VN document representation). Given the original term frequency, $c(w, d)$, the verbosity normalization is to obtain $k \cdot c(w, d)/v(d)$, where k is a verbosity scaling parameter. As a result, the verbosities of all VN document representations are fixed to k. Once all term frequencies in d are verbosity-normalized, the length of the resulting VN document representation becomes the scope; i.e., $k \cdot s(d)$, using Eq. (3).

$$k \sum_{w \in V} \frac{c(w, d)}{v(d)} = k \cdot s(d) \tag{3}$$

In scope normalization, we straightforwardly apply an existing retrieval model to the VN document representation. This can be regarded as a relaxed type of penalization because existing retrieval models perform relaxed penalization on document length.

In sum, let $f(d, q)$ be the original retrieval function that gives a score to d for query q. Let $\phi(d)$ be the VN representation of d, where $c(w, d)$ is replaced with $k \cdot c(w, d)/v(d)$ in all terms in d. Then, applying the two-stage normalization to $f(d, q)$ gives $f(\phi(d), q)$. Henceforth, we call $f(\phi(d), q)$ a VN retrieval model or a VN scoring function.

3.2. Generalization of two-stage document length normalization to pseudo relevance feedback

The generalization of the two-stage document length normalization to PRF is straightforward; for all the components of the PRF, we use VN document representations instead of original document representations. More specifically, an initial retrieval is first performed using the VN document representations. A new query model is then estimated from the top-retrieved documents using their relevance scores and their VN document representations. The VN document representations are again used for second-stage retrieval, which employs the estimated new query models.

4. Verbosity normalized pseudo-Relevance feedback

In this section, we present our process of applying the generalized two-stage normalization to RM and LCE to obtain VN-RM and VN-LCE, respectively.

4.1. Verbosity-normalized Markov random field

Before presenting LCE, we briefly review VN-MRF used in (Na, 2015) and derives VN-RM3.

MRFs are undirected graphical models that are used to define joint distributions over a set of random variables. To formally present the ranking function of the sequential dependence, suppose that q is a sequence of m terms $q_1 \cdots q_m$. According to the original framework, the relevance score of a document d is given by Metzler and Croft (2005)

$$f(d, q) = \lambda_T \sum_{q_i \in q} f_T(d, q_i) + \lambda_O \sum_{q_i q_{i+1} \in q} f_O(d, q_i q_{i+1}) + \lambda_U \sum_{q_i q_{i+1} \in q} f_U(d, q_i q_{i+1}) \tag{4}$$

where we have the constraint $\lambda_T + \lambda_O + \lambda_U = 1$, and $f_T(d, q_i)$, $f_O(d, q_i q_{i+1})$ and $f_U(d, q_i q_{i+1})$ are called the *feature functions* of the term, *ordered phrase*, and *unordered phrases*, respectively. Table 1 presents the definition of each feature function (Metzler & Croft, 2005).

To derive a VN retrieval model $f(\phi(d), q)$ for MRF, we replace the original term frequencies with the verbosity-normalized ones. As derived in (Na, 2015), Table 2 shows the verbosity-normalized feature functions used in the VN-MRF model. Unless stated otherwise, perplexity (defined in Eq. (15)) is assumed to be used for scope measure $s(d)$ due to its high effectiveness.

4.2. Verbosity normalized relevance model

RM3 (Abdul-jaleel et al., 2004; Lv & Zhai, 2009b) is a state-of-the-art variant of the relevance model (Lavrenko & Croft, 2001). According to Lavrenko (2004)'s generative view of relevance, queries (or relevant documents) are assumed to be random samples for the same underlying generative model, called the *relevance model*. Given q, let $p_{RM}(w|q)$ be the relevance model for the information need of q and $\mathscr{D}_{init}$ denotes the set of the initially retrieved documents in response to q. RM3 estimates $p_{RM}(w|q)$ using a posterior model over the document models in $\mathscr{D}_{init}$, conditioned on having observed sample q, resulting in

$$p_{RM}(w|q) = \sum_{d \in \mathscr{D}_{init}} p(w|d) p(d|q) \tag{5}$$

where $p(w|d)$ indicates the document model either with standard Dirichlet-prior smoothing (DP) of (Zhai & Lafferty, 2001a) or the verbosity normalized Dirichlet-prior smoothing (VN-DP) of (Na, 2015), and $p(d|q)$ is the posterior probability of document model $p(\cdot|d)$ after observing sample q, formulated as follows:

$$p(d|q) = \frac{p(d) \prod_{w \in q} p(w|d)^{c(w,q)}}{\sum_{d \in \mathscr{D}_{init}} p(d) \prod_{w \in q} p(w|d)^{c(w,q)}} \tag{6}$$

where $p(d)$ is the prior probability, which is assumed to be uniform.

Furthermore, the estimated relevance model over $\mathscr{D}_{init}$ is further interpolated with the original query model, ultimately resulting in the *expanded relevance model*, denoted by $p_{RM3}(w|q)$ as follows:

$$p_{RM3}(w|q) = (1 - \alpha) p_{ml}(w|q) + \alpha p_{RM}(w|q) \tag{7}$$

where $p_{ml}(w|q)$ indicates the MLE of the original query, computed by $c(w, q)/|q|$, and α is an interpolation parameter for controlling the combined weight of $p_{ml}(w|q)$ and $p_{RM}(w|q)$.

After $p_{RM3}(w|q)$ is estimated based on Eq. (7), document d is then re-scored by using the negative KL divergence between $p(w|q)$ and $p(w|d)$ (Lafferty & Zhai, 2001; Zhai & Lafferty, 2006). More generally, we use an additional smoothing parameter μ_F for second-stage retrieval, unlike in the case of the initial retrieval.

Eq. (7) is the common formula for referring to either RM3 or VN-RM3. In other words, Eq. (7) becomes RM3 if we use DP for $p(w|d)$ in Eq. (7), and it becomes VN-RM3 if we use VN-DP for $p(w|d)$. Specifically, RM and VN-RM correspond to the special cases of RM3 and VN-RM3, respectively, when $\alpha = 1$.

4.3. Verbosity-normalized latent concept expansion

Our formulation of VN-LCE is based on LCE. LCE assumes that, when users formulate an original query, they have some set of concepts in mind; however, they are only able to express a small number of them in the form of a query (Metzler & Croft, 2007). The concepts that the user has in mind but does not express in the query are called *latent concepts*. Latent concepts can be viewed as expansion terms in traditional terminology. More generally, they can consist of a single term, multiple terms, or a combination of single and multiple terms.

LCE is a process that is used to recover these latent concepts, in a manner similar to performing query expansion in traditional IR. In this study, LCE is used for conducting query expansion via the pseudo-relevance feedback. LCE first extends the original graph, by including original query terms, *expansion concepts* that we are interested in generating, and the document node. There are two variants for forming expansion concepts (Metzler & Croft, 2007) – *single term concept* and *multiple term concept* – depending on whether a dependency appears among expansion terms. We use the single-term concept in which no dependency relation is assumed among expansion terms, because it was shown to be the most effective despite the simplicity. Fig. 1 illustrates an example of the extended graph in which a single-term concept (e_1) is appended to the original graph consisting of a three-word query ($q = q_1 q_2 q_3$).

Following the original LCE approach, after the extended graph is constructed, we compute $p(e|q)$, the conditional probability of a latent concept given the query, according to

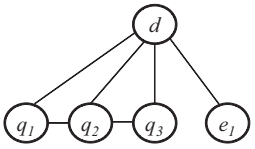

Fig. 1. Graphical model representation for LCE using single-term concepts for a three-term query (Metzler and Croft, 2007).

$$p(e|q) \propto \sum_{d \in \mathscr{D}_{init}} \exp\Bigg(\lambda_T \sum_{q_i \in q} f_T(d, q_i) + \lambda_O \sum_{q_i q_{i+1} \in q} f_O(d, q_i q_{i+1}) + \lambda_U \sum_{q_i q_{i+1} \in q} f_U(d, q_i q_{i+1}) + \lambda_E f_E(d, e) + \lambda_C f_C(e) \Bigg) \tag{8}$$

In addition to the original feature functions of MRF, we have $f_E(d, e)$ and $f_C(e)$ as new feature functions for expansion concept e, as defined in Table 3.

Using the definitions of all feature functions listed in Table 1 and 3, the complete form of Eq. (8) is given as:

$$\begin{aligned} p(e|q) \propto \sum_{d \in \mathscr{D}_{init}} \exp \Bigg(& \lambda_T \sum_{q_i \in q} \ln\left[(1-\gamma_d^T) \frac{c(q_i, d)}{|d|} + \gamma_d^T \frac{c(q_i, C)}{|C|} \right] \\ & + \lambda_O \sum_{q_i q_{i+1} \in q} \ln\left[(1-\gamma_d^T) \frac{c_{\#1}(q_i q_{i+1}, d)}{|d|} + \gamma_d^T \frac{c_{\#1}(q_i q_{i+1}, C)}{|C|} \right] \\ & + \lambda_U \sum_{q_i q_{i+1} \in q} \ln\left[(1-\gamma_d^T) \frac{c_{\#uw8}(q_i q_{i+1}, d)}{|d|} + \gamma_d^T \frac{c_{\#uw8}(q_i q_{i+1}, C)}{|C|} \right] \\ & + \lambda_E \ln\left[(1-\gamma_d^E) \frac{c(e, d)}{|d|} + \gamma_d^E \frac{c(e, C)}{|C|} \right] \\ & + \lambda_C \ln\left[\frac{c(e, C)}{|C|} \right] \Bigg) \end{aligned} \tag{9}$$

where γ_d^T is $\frac{\mu_T}{\mu_T + |d|}$ and γ_d^E is $\frac{\mu_E}{\mu_E + |d|}$. This separate use of smoothing parameters for γ_d^T and γ_d^E allows us to optimize a smoothing parameter according to whether the model of interest is a *query model* or a *document model;* μ_E is targeted for estimating a query model, and μ_T for a document model. This property is unlike the case of RM3, in which the same smoothing parameter value is used for both query and document models.

Based on Eq. (9), we select q_E, the set of n latent concepts with the highest likelihood of $p(e|q)$. A new graph is constructed by augmenting the original graph with n expansion concepts, $q_E = e_1 \cdots e_n$. However, when augmenting the original graph, following the original LCE approach, we further assume that the expansion concepts of q_E have a different degree of importance from the original query terms of q, controlled by an additional parameter α. As a result, documents are ranked according to $f(d, q\, q_E)$ using Eq. (4) with the augmented graph, which is equivalently rewritten as:

Table 1
Feature functions used in the MRF model. $c_{\#1}(q_i q_{i+1}, d)$ indicates the number of times that the *exact phrase* $q_i q_{i+1}$ occurs in document d, and $c_{\#un8}(q_i q_{i+1}, d)$ indicates the number of times that both terms q_i and q_{i+1} appear *ordered* or *unordered* within a window with a span of 8.

Feature	Value
$f_T(d, q_i)$	$\ln\left[\frac{c(q_i, d) + \mu_T \frac{c(q_i, C)}{\|C\|}}{\|d\| + \mu_T} \right]$
$f_O(d, q_i q_{i+1})$	$\ln\left[\frac{c_{\#1}(q_i q_{i+1}, d) + \mu_O \frac{c_{\#1}(q_i q_{i+1}, C)}{\|C\|}}{\|d\| + \mu_O} \right]$
$f_U(d, q_i q_{i+1})$	$\ln\left[\frac{c_{\#un8}(q_i q_{i+1}, d) + \mu_U \frac{c_{\#un8}(q_i q_{i+1}, C)}{\|C\|}}{\|d\| + \mu_U} \right]$

Table 2
Verbosity-normalized feature functions used in the VN-MRF model where $f_T(\phi(d), q_i)$, $f_O(\phi(d), q_i q_{i+1})$, and $f_U(\phi(d), q_i q_{i+1})$ are verbosity-normalized feature functions that correspond to original feature functions.

Feature	Value
$f_T(\phi(d), q_i)$	$\ln\left[\frac{\frac{c(q_i, d)}{v(d)} + \mu_T \frac{c(q_i, C)}{\lvert C\rvert}}{s(d) + \mu_T}\right]$
$f_O(\phi(d), q_i q_{i+1})$	$\ln\left[\frac{\frac{c_{\#1}(q_i q_{i+1}, d)}{v(d)} + \mu_O \frac{c_{\#1}(q_i q_{i+1}, C)}{\lvert C\rvert}}{s(d) + \mu_O}\right]$
$f_U(\phi(d), q_i q_{i+1})$	$\ln\left[\frac{\frac{c_{\#un8}(q_i q_{i+1}, d)}{v(d)} + \mu_U \frac{c_{\#un8}(q_i q_{i+1}, C)}{\lvert C\rvert}}{s(d) + \mu_U}\right]$

Table 3
Additional feature functions for LCE.

Feature	Value
$f_E(d, e)$	$\log\left[\frac{c(e, d) + \mu_E \frac{c(e, C)}{\lvert C\rvert}}{\lvert d\rvert + \mu_E}\right]$
$f_C(e)$	$\log\left[\frac{c(e, C)}{\lvert C\rvert}\right]$

$$
\begin{aligned}
f(d, q\ q_E) \;=\; & (1-\alpha)\left(\lambda_T \sum_{q_i \in q} f_T(d, q_i) + \lambda_O \sum_{q_i q_{i+1} \in q} f_O(d, q_i q_{i+1}) + \lambda_U \sum_{q_i q_{i+1} \in q} f_U(d, q_i q_{i+1})\right) \\
& + \alpha\left(\lambda_T \sum_{e \in q_E} p(e|q) f_E(d, e)\right)
\end{aligned} \tag{10}
$$

From the definitions of the feature functions in Table 1 and 3, we obtain the following complete form of Eq. (10):

$$
\begin{aligned}
f(d, q\ q_E) \;=\; & (1-\alpha)\left(\lambda_T \sum_{q_i \in q} \ln\left[(1-\gamma_d^T)\frac{c(q_i, d)}{\lvert d\rvert} + \gamma_d^T \frac{c(q_i, C)}{\lvert C\rvert}\right]\right. \\
& + \lambda_O \sum_{q_i q_{i+1} \in q} \ln\left[(1-\gamma_d^T)\frac{c_{\#1}(q_i q_{i+1}, d)}{\lvert d\rvert} + \gamma_d^T \frac{c_{\#1}(q_i q_{i+1}, C)}{\lvert C\rvert}\right] \\
& \left. + \lambda_U \sum_{q_i q_{i+1} \in q} \ln\left[(1-\gamma_d^T)\frac{c_{\#uw8}(q_i q_{i+1}, d)}{\lvert d\rvert} + \gamma_d^T \frac{c_{\#uw8}(q_i q_{i+1}, C)}{\lvert C\rvert}\right]\right) \\
& + \alpha\left(\lambda_T \sum_{e \in q_E} p(e|q) \ln\left[(1-\gamma_d^T)\frac{c(e, d)}{\lvert d\rvert} + \gamma_d^T \frac{c(e, C)}{\lvert C\rvert}\right]\right)
\end{aligned} \tag{11}
$$

where $p(e|q)$ indicates Eq. (9).

It is trivial to show that LCE using Eq. (10) includes RM3 as a special case (Metzler & Croft, 2007); Eq. (10) becomes RM3 when $\lambda_T = 1$, $\lambda_E = 1$, $\lambda = 0$, and $\mu_E = \mu_T$. More generally, we use a new smoothing parameter μ_F to compute $f_T(q_i, d)$ at the second-stage retrieval; this parameter is different from μ_T, the one used at the initial retrieval.

We now formulate VN-LCE by applying the generalized two-stage normalization to all processing steps in the pseudo-relevance feedback – initial retrieval, the selection of expansion concepts, and second-stage retrieval. First, for selecting expansion terms, we use the modified formula of Eq. (8), given as:

Table 4
Additional verbosity normalized feature functions for VN-LCE.

Feature	Value
$f_E(\phi(d), e)$	$\log\left[\frac{\frac{c(e,d)}{v(d)}+\mu_E\frac{c(e,C)}{\lvert C\rvert}}{s(d)+\mu_E}\right]$

$$p(e|q) \propto \sum_{d\in\mathscr{D}_{init}} \exp\Bigg(\lambda_T \sum_{q_i\in q} f_T(\phi(d), q_i) + \lambda_O \sum_{q_iq_{i+1}\in q} f_O(\phi(d), q_iq_{i+1}) + \lambda_U \sum_{q_iq_{i+1}\in q} f_U(\phi(d), q_iq_{i+1}) + \lambda_E f_E(\phi(d), e) + \lambda_C f_C(e)\Bigg) \tag{12}$$

where $f_E(\phi(d), e)$ is the verbosity-normalized feature function corresponding to $f_E(d, e)$, as defined in the Table 4.

The complete form of Eq. (12) is the same as that of Eq. (8), except that VN-LCE uses $\gamma_d^T = \frac{v(d)\mu_T}{|d|+v(d)\mu_T}$ and $\gamma_d^E = \frac{v(d)\mu_E}{|d|+v(d)\mu_E}$.

We then substitute the VN feature functions $f_T(q_i, \phi(d))$, $f_O(q_i, \phi(d))$, and $f_U(q_i, \phi(d))$ for feature functions in Eq. (10), and finally obtain the following VN model $f(\phi(d), q\ q_E)$:

$$f(\phi(d), q\ \ q_E) = (1-\alpha)\Bigg(\lambda_T \sum_{q_i\in q} f_T(q_i, \phi(d)) + \lambda_O \sum_{q_iq_{i+1}\in q} f_O(\phi(d), q_iq_{i+1}) + \lambda_U \sum_{q_iq_{i+1}\in q} f_U(\phi(d), q_iq_{i+1})\Bigg) + \alpha\Bigg(\lambda_T \sum_{e\in q_E} p(e|q) f_E(\phi(d), e)\Bigg) \tag{13}$$

Again, the complete form of Eq. (13) is the same as that of Eq. (11), except that VN-LCE uses $\gamma_d^T = \frac{v(d)\mu_T}{|d|+v(d)\mu_T}$ and $\gamma_d^E = \frac{v(d)\mu_E}{|d|+v(d)\mu_E}$.

5. Analysis on verbosity-preserving and verbosity-sensitive effects on a feedback query model

In this section, we compare two PRF methods viz., RM and VN-RM by focusing on the difference in their verbosity-preserving and verbosity-sensitive effects on feedback query models[8].

According to Sections 3–4, the main differences between RM and VN-RM are summarized as follows:

1. A set of feedback documents ($\mathscr{D}_{init}$): RM uses the top-retrieved documents from DP, whereas VN-RM uses ones from VN-DP.
2. The estimation of a feedback query model ($p_{RM}(w|q)$): In Eq. (5), RM uses the document model of DP for $p(w|d)$ (i.e., $\frac{c(w,d)+\mu p(w|C)}{|d|+\mu}$), whereas VN-RM uses the verbosity normalized document model of VN-DP (i.e., $\frac{c(w,d)+\mu v(d)p(w|C)}{|d|+\mu v(d)}$) as in Na (2015).
3. The second-stage retrieval model ($f(q, d)$): RM uses the document model with DP, whereas VN-RM uses the document model with VN-DP – $f(q, \phi(d))$.

Among these three differences, we address only the first two (viz., $\mathscr{D}_{init}$ and $p_{RM}(w|q)$), because they are related to the verbosity-preserving and verbosity-sensitive effects, respectively[9].

The main results of our analysis are summarized as follows:

1. Verbosity-preserving effects of RM and VN-RM: According to Na (2015), RM tends to prefer verbose documents to normal ones, thus resulting in high verbosities in feedback documents and accordingly in the feedback query model. On the other hand, VN-RM strictly penalizes verbose documents, thus enabling it to return *less verbose* feedback documents during the initial-stage retrieval. Accordingly, the query verbosity of the resulting feedback query model will tend to be lower than that of an RM-based query model.
2. Verbosity-sensitive effects of RM and VN-RM: RM is highly sensitive to verbose documents in estimating a feedback query model, whereas VN-RM normalizes the verbosities of documents and significantly reduces the sensitivity to verbose documents.

In the following subsections, we discuss these results in more detail.

[8] Because the MRF features used in LCE and VN-LCE in this paper are mainly from the generative probabilities $p(w|d)$ used in RM and VN-RM, respectively, we do not compare LCE and VN-LCE separately. This is because the analysis holds similarly between LCE and VN-LCE.

[9] For the third difference, as discussed in Na (2015), we empirically showed that VN-DP was more effective than DP, especially for verbose queries. Given the results, it is expected that VN-RM can be more effective than RM when the verbosity of a feedback query model is relatively high.

Table 5
Average verbosity (*avgv*) and average length (*avgl*) of top-retrieved documents for DP and VN-DP in ROBUST, WT10G, GOV2, and ClueWeb6 (Cat-B) collections, where R is selected from among {5, 10, 15, 20, 30}.

R	Method	ROBUST		WT10G		GOV2		ClueWeb09	
		avgv	*avgl*	*avgv*	*avgl*	*avgv*	*avgl*	*avgv*	*avgl*
$R = 5$	DP	2.88	706.6	6.58	1491.0	9.45	2641.9	22.59	3273.7
	VN-DP	2.43	603.0	3.64	1352.9	6.64	3202.1	10.84	3346.9
$R = 10$	DP	2.74	666.5	5.49	1300.9	8.89	2541.5	18.88	2658.5
	VN-DP	2.42	584.2	3.45	1202.6	6.44	3030.5	8.97	2857.1
$R = 15$	DP	2.66	634.6	5.15	1311.3	8.90	2607.7	17.02	2409.8
	VN-DP	2.38	562.0	3.41	1156.3	6.23	2881.1	8.02	2571.5
$R = 20$	DP	2.63	624.6	5.10	1296.4	8.80	2639.6	15.79	2250.0
	VN-DP	2.37	553.0	3.43	1159.6	6.13	2780.4	7.51	2421.1
$R = 30$	DP	2.61	617.4	5.26	1238.3	8.30	2628.2	14.02	2023.1
	VN-DP	2.33	534.3	3.42	1151.9	6.12	2758.6	6.70	2215.4

5.1. Verbosity-preserving effect: Based on the verbosity of feedback documents.

Na (2015) shows analytically that DP likely prefers verbose documents to normal ones, whereas VN-DP strictly penalizes verbose documents. As a result, it is expected that DP will tend to place more verbose documents in the top-retrieved documents, and VN-DP will tend to return less-verbose ones.

To empirically verify whether DP indeed prefers verbose documents to normal ones and VN-DP strictly penalizes verbose ones, Table 5 compares the average verbosity (*avgv*) and the average length (*avgl*) of top-retrieved documents for DP and VN-DP in the collections, ROBUST, WT10G, GOV2, and ClueWeb6 (Cat-B)[10] where R is selected from among {5, 10, 15, 20, 30} and the perplexity (defined in Eq. (15)) is used to measure the scope $s(d)$.

As shown in Table 5, the average verbosities of top-retrieved documents by VN-DP are lower than those by DP for all the test collections and values of R. The difference in the average verbosity between VN-DP and DP depends on the test collection. With ROBUST, for instance, the average verbosities resulting from VN-DP are only slightly lower than those from DP. With ClueWeb09, by contrast, the average verbosities from VN-DP are approximately half what they are from DP.

In addition, the average lengths resulting from VN-DP are not consistently lower than those from DP for all test collections. In ROBUST and WT10G, the average lengths of top-retrieved documents in VN-DP are lower than those in DP, whereas in GOV2 and ClueWeb09 the average lengths in VN-DP are higher than those in DP. Therefore, unlike the document verbosity, the document length is not a consistent metric for capturing the difference between top-retrieved documents in VN-DP and DP.

5.2. Verbosity-sensitive effect

To discuss verbosity-sensitive effects formally, we first introduce a type of perturbation, called *C-type* (i.e., copying perturbation), defined as follows:

• **C-type perturbation**: The operator $\psi(\cdot)$ is called *C-type* if for $d_1 = \psi(d_2)$, $c(w, d_2) = K \cdot c(w, d_1)$ (i.e., $w \in d_2$) for all $w \in V$, where K is a constant value.

A C-type perturbation is a verbosity-increasing operator by maintaining all the content in an original document such that it is exactly the same after the perturbation.

One of the verbosity-sensitive effects is formally captured by checking whether a PRF method violates **VNC-QM** (i.e., verbosity normalization preference for the query model), defined as follows:

• **VNC-QM**: Suppose that $TW(w, d)$ be the weight of the expansion term w of document d of a PRF method. If $d_1 = \psi(d_2)$ and $\psi(\cdot)$ is a C-type perturbation, then $TW(d_1, q) = TW(d_2, q)$ for $K > 1$.

In VNC-QM, it should be noted that $TW(w, d)$ is the component that corresponds to the weight of the expansion term obtained without using an original query.

The following theorem states that the feedback query models resulting from RM and VN-RM differ in terms of satisfying VNC-QM:

Theorem 1. *Let $TW_{RM}(w, d)$ and $TW_{VN\text{-}RM}(w, d)$ denote $TW(w, d)$, as used in RM and VN-RM, respectively, and defined as follows:*

$$TW_{RM}(w, d) = \frac{c(w, d) + \mu p(w|C)}{|d| + \mu}$$

$$TW_{VN\text{-}RM}(w, d) = \frac{c(w, d) + \mu v(d) p(w|C)}{|d| + \mu v(d)}$$

Then, VN-RM satisfies VNC-QM, whereas RM does not.

[10] A detailed description of the collections is provided in Table 7

Because it is trivial to prove this theorem, we instead introduce a simple example to illustrate the key idea in the theorem:

• *Example to check whether RM and VN-RM satisfies VNC-QM*: Suppose that we use UniqLength as the scope measure, and g and h denote terms. Our example is given as two documents, d_1 and d_2, that consist exclusively of g and h, where d_1 is the document obtained by applying a C-type perturbation to d_2 with $K = 2$, as follows:

$$\boxed{\begin{array}{l} d_1 = g\ g\ h\ h \\ d_2 = g\ h \end{array}}$$

where $v(d_1) = 2$ and $v(d_2) = 1$, under the definition of verbosity.

In these example documents, RM and VN-RM differ in terms of satisfying VNC-QM.

1) RM: $TW(g, d_1) > TW(g, d_2)$, i.e. RM does not satisfy VNC-QM, as shown in the following:

$$\begin{aligned} TW_{RM}(w, d_1) &= \frac{c(w, d_1) + \mu p(w|C)}{|d_1| + \mu} = \frac{2{\cdot}c(w, d_2) + \mu p(w|C)}{2{\cdot}|d_2| + \mu} \\ TW_{RM}(w, d_2) &= \frac{c(w, d_2) + \mu p(w|C)}{|d_2| + \mu} \end{aligned}$$

2) VN-RM: $TW(g, d_1) = TW(g, d_2)$, i.e. VN-RM satisfies VNC-QM, as derived from the following:

$$\begin{aligned} TW_{VN\text{-}RM}(w, d_1) &= \frac{c(w, d_1) + \mu v(d_1) p(w|C)}{|d_1| + \mu v(d_1)} = \frac{2{\cdot}c(w, d_2) + \mu{\cdot}2{\cdot}v(d_2) p(w|C)}{2{\cdot}|d_2| + \mu{\cdot}2{\cdot}v(d_2)} \\ &= \frac{c(w, d_2) + \mu v(d_2) p(w|C)}{|d_2| + \mu v(d_2)} \\ TW_{VN\text{-}RM}(w, d_2) &= \frac{c(w, d_2) + \mu v(d_2) p(w|C)}{|d_2| + \mu v(d_2)} \end{aligned}$$

□.

Furthermore, let $p_{RM}(w|q, \mathscr{D}_{init})$ and $p_{VN\text{-}RM}(w|q, \mathscr{D}_{init})$ denote the feedback query models that result from applying RM and VN-RM on $\mathscr{D}_{init}$, respectively. Here, $p_{RM}(w|q)$ and $p_{VN\text{-}RM}(w|q)$ are rewritten in terms of $TW_{RM}(w, d)$ and $TW_{VN\text{-}RM}(w, d)$, respectively, as follows:

$$\left.\begin{aligned} p_{RM}(w|q) &\propto \sum_{d \in D_{init}} TW_{RM}(w, d) \exp(f(q, d)) \\ p_{VN\text{-}RM}(w|q) &\propto \sum_{d \in D_{init}} TW_{VN\text{-}RM}(w, d) \exp(f(q, \phi(d))) \end{aligned}\right) \tag{14}$$

where $f(q, d)$ and $f(q, \phi(d))$ respectively denote the original and the verbosity normalized scoring functions of d to the original query q.

Under this definition of the feedback query models in Eq. (14), we infer that the verbosity-sensitive effect on a feedback query model is more pronounced in RM than in VN-RM. Again, suppose that d_1 is the document obtained by applying the C-type perturbation to d_2. In RM, $f(q, d_1) > f(q, d_2)$. Thus, $p_{RM}(w|q)$ gives more weight to verbose documents than to normal ones, making the resulting query model highly sensitive to the verbosity of feedback documents. In VN-RM, $f(q, \phi(d_1)) = f(q, \phi(d_2))$. Therefore, $p_{VN\text{-}RM}(w|q)$ is not sensitive to the verbosity of feedback documents.

In sum, Table 6 shows the differences in the verbosity-sensitive effects of the RM and VN-RM on feedback query models under copy perturbation, when $d_1 = \psi(d_2)$ and $\psi(\cdot)$ is C-type. As shown in the table, where VN-RM is not sensitive to verbose documents, RM is highly sensitive to verbose documents, in terms of both the weight of the expansion term and that of the feedback document.

6. Experimental setting

6.1. Experimental setup

For evaluation, we used three different standard TREC collections – ROBUST, WT10G, GOV2, and ClueWeb09 (Category B).

Table 6
Differences in the verbosity-sensitive effects of the RM and VN-RM on feedback query models, when $d_1 = \psi(d_2)$ and $\psi(\cdot)$ is C-type.

	VN-RM	RM
Expansion term weight ($TW(w, d)$)	$TW(w, d_1) = TW(w, d_2)$	$TW(w, d_1) > TW(w, d_2)$
Feedback document weight ($f(q, d)$)	$f(w, \phi(d_1)) = f(w, \phi(d_2))$	$f(w, d_1) > f(w, d_2)$
Feedback query model	$p_{RM}(w\|q, \{d_1\} \cup \mathscr{F})$	$p_{RM}(w\|q, \{d_1\} \cup \mathscr{F})$
($p_{RM}(w\|q, \mathscr{D}_{init})$)	$= p_{RM}(w\|q, \{d_2\} \cup \mathscr{F})$	$> p_{RM}(w\|q, \{d_2\} \cup \mathscr{F})$
VNC-QM	yes	no

Table 7
Statistics of each test collection.

Statistics	ROBUST	WT10G	GOV2	ClueWeb09 (Cat-B)
NumDocs	528,156	1,692,096	25,205,179	50,220,423
NumWords	572,180	6,346,858	40,002,579	148,705,592
TopicSet	Q301 - 450 Q601 - 700	Q451 - 550	Q701 - 850	Q1 - 200

Table 7 lists the basic statistics of each test collection, where *NumDocs* is the number of documents, *NumWords* is the total number of word occurrences in each collection, *TopicSet* is the range of topic numbers used for training and testing.

All experiments were performed using the Lemur toolkit (version 4.12). We carried out standard preprocessing by applying the Porter stemmer and removing stopwords from the standard INQUERY stoplist (Allan et al., 2000)[11]. The short keywords (titles) of TREC topics were used for queries at the initial retrieval and the second retrieval stages.

For the evaluation, we used the mean average precision (MAP) (Croft et al., 2009) and a normalized discounted cumulative gain at rank 20 (NDCG@20) (Järvelin & Kekäläinen, 2002).

For each query, our evaluation is based on the top 1000 documents retrieved. We also report significance test results by a non-directional paired *t*-test at 0.95 confidence level. For the significance test, we use all *per-topic* performances in a collection, i.e., the number of performance difference samples used for the *t*-test is the same as the total number of topics in a given collection

Given a test topic set consisting of 50 queries, each parameter was tuned using the other topic sets in the same test collection as the development set[12].

6.2. Scope measures

Among variants of scope measure in Na (2015), we use perplexity due to its high effectiveness. Under perplexity, $s(d)$ is defined as follows:

$$s(d) = \begin{cases} \exp(-\sum_w p_{ml}(w|d) ln(p_{ml}(w|d))) & if |d| \geq 1 \\ 0 & otherwise \end{cases} \quad (15)$$

where $p_{ml}(w|d)$ is defined by $c(w, d)/|d|$, which is the maximum likelihood estimation (MLE) of the document language model for d.

6.3. Parameter tuning

For each feedback method, Table 8 summarizes what initial retrieval model is used for obtaining $\mathscr{D}_{init}$, and how retrieval parameters are determined or trained.

As in Na (2015), we use the following search space for training the retrieval parameters μ and α as follows:

- μ: { 100, 200, 300, 400, 500, 600, 800, 1000, 1500, 2000, 2500, 3000, 4000, 5000, 7000, 10000, 15000, 20000 }
- α: { 0.1, 0.25, 0.5, 0.75, 0.9, 0.95, 1.0 }

By adopting the setting used in the recent work on RM3 (Lv & Zhai, 2009b), μ_F was fixed to 1000 for all four methods. In addition, we fixed $\lambda_T = 0.85$, $\lambda_O = 0.1$ for MRF and LCE (and VN-MRF and VN-LCE), as used in Na (2015). The parameter μ_E was fixed to 1 based on the sample experimental setting of the Ivory Toolkit on ROBUST[13]. To further reduce the parameter search cost, we did not search λ_E and λ_C and instead fixed them using $\lambda_E = 1.0$ and $\lambda_C = 0.0$. This setting of λ_E and λ_C allows us to minimally modify MRF so as to be closer to RM3, without significant performance degradation[14].

7. Experimental results

In this section, we present the performances of the four different methods – RM3, VN-RM3, LCE and VN-LCE – for pseudo-

[11] Depending on collection, we often need to include one or two stopwords from the stoplist, in order to handle queries consisting of only stopwords. An example query is Q531: “who and whom”.

[12] As in Na (2015), a topic set comprises 50 queries, created in each year of TREC. For example, in ROBUST, as 250 queries are available, there are 5 topic sets: namely, TREC6(Q301 - Q350), TREC7(Q351 - Q400), TREC8(Q401 - Q450), ROBSUT03(Q601 - Q650), and ROBUST04(Q651 - Q700). The parameters used when testing the 50 queries in each topic set were trained using the remaining 200 queries in other topic sets as training data. In other words, to test 50 queries in TREC6, queries Q351 - Q450 and Q601 - Q700 in TREC7, TREC8, ROBUST03, and ROBUST04 were used as training data. To test queries in TREC7, queries in TREC6, TREC8, ROBUST03, and ROBUST04 were used as the training set, and so on. Therefore, for ROBUST, we used five-fold cross validation for parameter-tuning, where the folds were fixed. For WT10G, where 100 queries are used, we have two topic sets, namely TREC9(Q451 - Q500) and TREC10(Q501 - Q550). To test 50 queries in TREC9, we used queries in TREC10 as the training set, and vice-versa. Thus, for WT10G, we used two-fold cross validation for parameter-tuning. Similarly, we used three- and four-fold cross validation for GOV2 and ClueWeb09, respectively.

[13] https://lintool.github.io/Ivory/

[14] In our preliminary experiment, we also evaluated LCE using the setting $\mu_E = 0.7$ and $\mu_C = 0.15$ on ROBUST and WT10G, which is similar to those in the sample experiment of the Ivory toolkit. In this preliminary experiment, we observed that LCE led to slight improvements on ROBUST over the simplest case using $\mu_E = 1.0$ and $\mu_C = 0.0$, and it led to some performance degradation on WT10G.

Table 8
Summary of parameter selection for RM3, VN-RM3, LCE, and VN-LCE. "Trained" indicates that a given retrieval parameter is trained over the training sets, instead of being fixed to a specific value.

Method	$\mathscr{D}_{init}$	Parameter
RM3	DP	μ, α: Trained
VN-RM3	VN-DP	$\mu_F = 1{,}000$
LCE	MRF	μ, α: Trained
VN-LCE	VN-MRF	$\lambda_T = 0.85$, $\lambda_D = 1.0$ $\lambda_E = 1.0$, $\lambda_C = 0.0$ $\mu_E = 1$, $\mu_F = 1{,}000$

relevance feedback. Our implementation of LCE is based on the version distributed by the Ivory Toolkit, in which the full implementations of MRF and LCE are available.

7.1. VN-RM3 Vs. RM3

Table 9 shows the comparison results of the four methods on the three test collections, where R is selected from among 5, 10, 15, 20, and 30. The last row of Table 9 indicates the result for the same setting, but with the value of R trained based on the training set. For all PRF methods, the number of expansion terms is fixed to 100.

It can be seen that VN-RM3 improves RM3 slightly on ROBUST, and significantly on WT10G, GOV2, and ClueWeb09. Specifically, in the Web collections, VN-RM3 is clearly better than RM3, achieving about 2%, 3%, and 2–3% increases in MAP of RM3 on WT10G, GOV2, and ClueWeb09, respectively, with most improvements being statistically significant. On ROBUST, VN-RM3 only slightly improves over RM3, showing an approximately 0.5% increase in MAP, but it is still statistically significant when R is trained (i.e., the last row).

The overall comparison results of VN-RM3 over RM3 are correlated with the differences of verbosities of feedback documents between VN-DP and DP shown in Table 5. The WT10G, GOV2, and ClueWeb09 collections are the most effective collections in that the significant differences in verbosities of feedback documents between VN-DP and DP are observed.

7.2. VN-LCE Vs. LCE

Table 9 clearly shows that VN-LCE is better than LCE, with statistically significant improvements in all test collections and most

Table 9
MAP performance comparison of four pseudo-relevance feedback methods RM3, VN-RM3, LCE, and VN-LCE. MAP is used as the evaluation metric. The symbols α, β, and γ indicate statistically significant improvements at 0.95 confidence level over DP, RM3, and LCE, respectively. The number of expansion terms is 100.

R		ROBUST	WT10G	GOV2	ClueWeb09
Init	DP	0.2447	0.1963	0.2907	0.1228
	MRF	0.2545^{α}	0.2149^{α}	0.3095^{α}	0.1312^{α}
$R = 5$	RM3	0.2837^{α}	0.2229^{α}	0.3085^{α}	0.1238
	VN-RM3	0.2845^{α}	$0.2418^{\alpha\beta}$	$0.3392^{\alpha\beta}$	$0.1484^{\alpha\beta}$
	LCE	$\mathbf{0.2955}^{\alpha\beta}$	0.2296^{α}	$0.3321^{\alpha\beta}$	$0.1317^{\alpha\beta}$
	VN-LCE	$0.2944^{\alpha\beta}$	$\mathbf{0.2603}^{\alpha\beta\gamma}$	$\mathbf{0.3698}^{\alpha\beta\gamma}$	$\mathbf{0.1593}^{\alpha\beta\gamma}$
$R = 10$	RM3	0.2797^{α}	0.2182^{α}	0.3158^{α}	0.1263
	VN-RM3	$0.2890^{\alpha\beta}$	$0.2329^{\alpha\beta}$	$0.3441^{\alpha\beta}$	$0.1490^{\alpha\beta}$
	LCE	$0.2979^{\alpha\beta}$	0.2301^{α}	$0.3356^{\alpha\beta}$	$0.1370^{\alpha\beta}$
	VN-LCE	$\mathbf{0.3062}^{\alpha\beta\gamma}$	$\mathbf{0.2535}^{\alpha\beta\gamma}$	$\mathbf{0.3760}^{\alpha\beta\gamma}$	$\mathbf{0.1602}^{\alpha\beta\gamma}$
$R = 15$	RM3	0.2837^{α}	0.2129^{α}	0.3180^{α}	0.1285^{α}
	VN-RM3	0.2898^{α}	$0.2275^{\alpha\beta}$	$0.3463^{\alpha\beta}$	$0.1512^{\alpha\beta}$
	LCE	$0.2999^{\alpha\beta}$	0.2266^{α}	$0.3344^{\alpha\beta}$	$0.1382^{\alpha\beta}$
	VN-LCE	$\mathbf{0.3122}^{\alpha\beta\gamma}$	$\mathbf{0.2466}^{\alpha\beta\gamma}$	$\mathbf{0.3737}^{\alpha\beta\gamma}$	$\mathbf{0.1623}^{\alpha\beta\gamma}$
$R = 20$	RM3	0.2816^{α}	0.2113^{α}	0.3165^{α}	0.1271
	VN-RM3	0.2868^{α}	$0.2258^{\alpha\beta}$	$0.3440^{\alpha\beta}$	$0.1528^{\alpha\beta}$
	LCE	$0.3015^{\alpha\beta}$	0.2191^{α}	$0.3325^{\alpha\beta}$	$0.1397^{\alpha\beta}$
	VN-LCE	$\mathbf{0.3114}^{\alpha\beta\gamma}$	$\mathbf{0.2490}^{\alpha\beta\gamma}$	$\mathbf{0.3745}^{\alpha\beta\gamma}$	$\mathbf{0.1622}^{\alpha\beta\gamma}$
$R = 30$	RM3	0.2767^{α}	0.2060^{α}	0.3071^{α}	0.1279
	VN-RM3	$0.2827^{\alpha\beta}$	$0.2232^{\alpha\beta}$	$0.3455^{\alpha\beta}$	$0.1568^{\alpha\beta}$
	LCE	$0.3005^{\alpha\beta}$	0.2159^{α}	$0.3324^{\alpha\beta}$	$0.1413^{\alpha\beta}$
	VN-LCE	$\mathbf{0.3090}^{\alpha\beta\gamma}$	$\mathbf{0.2440}^{\alpha\beta\gamma}$	$\mathbf{0.3722}^{\alpha\beta\gamma}$	$\mathbf{0.1648}^{\alpha\beta\gamma}$
Trained	RM3	0.2813^{α}	0.2148^{α}	0.3153^{α}	0.1279
	VN-RM3	$0.2881^{\alpha\beta}$	$0.2341^{\alpha\beta}$	$0.3450^{\alpha\beta}$	$0.1568^{\alpha\beta}$
	LCE	$0.2999^{\alpha\beta}$	$0.2295^{\alpha\beta}$	$0.3341^{\alpha\beta}$	$0.1413^{\alpha\beta}$
	VN-LCE	$\mathbf{0.3122}^{\alpha\beta\gamma}$	$\mathbf{0.2603}^{\alpha\beta\gamma}$	$\mathbf{0.3736}^{\alpha\beta\gamma}$	$\mathbf{0.1648}^{\alpha\beta\gamma}$

Table 10
The MAP performances of LCE reported in Lang et al. [2010], relative to those of DP, MRF, and LCE_HMRF, where LCE_HMRF indicates the result of their proposed method. The symbols α, β, and γ indicate statistically significant improvements at 0.95 confidence level over DP, RM3, and LCE, respectively.

	ROBUST	WT10G	GOV2
DP	0.2532	0.1968	0.2981
MRF	0.2653^{α}	0.2073^{α}	0.3244^{α}
RM3	0.2834	0.2118	0.3179
LCE	$0.3057^{\alpha\beta}$	$0.2259^{\alpha\beta}$	$0.3313^{\alpha\beta}$
LCE_HMRF	$0.3313^{\alpha\beta\gamma}$	$0.2454^{\alpha\beta\gamma}$	$0.3634^{\alpha\beta\gamma}$
VN-LCE	0.3122	0.2603	0.3736

cases of *R*s. Specifically, in Web collections, the improvements using VN-LCE are much larger than those in ROBUST, with almost all cases being statistically significant, achieving approximately 3%, 4%, and 2–3% increases in MAP over LCE on WT10G, GOV2, and ClueWeb09, respectively. On ROBUST, despite the smaller amounts than in the case of Web collections, the improvements of LCE by using VN-LCE are almost all significant, but only except for $R = 5$.

When comparing the improvements using VN-RM3 and VN-LCE, it can be seen that VN-LCE tends to show slightly larger improvements of LCE than that of RM3 by VN-RM3 – for example, in ROBUST, VN-LCE results in small but consistently significant improvements of LCE, whereas VN-RM3 does not improve RM3 so much.

To further compare VN-LCE's performances with other works based on LCE, Table 10 presents the performances of MRF, RM3, and LCE, including that of LCE_HMRF, which was reported in Lang et al. (2010). LCE_HMRF is the work's suggested approach. Because *R* was trained in Lang et al. (2010), the performances of the methods listed in Table 10 are fairly comparable to those listed in the last row of Table 9. Looking at the improvement of RM3 using LCE, both results are highly similar, except that the performances of the baselines (DP and MRF) are slightly different between them.

Furthermore, on WT10G and GOV2, VN-LCE's performances (the last row of Table 9) are slightly better than those of LCE_HMRF. Specifically, when we look at how each method (VN-LCE and LCE_HMRF) improves LCE, VN-LCE shows slightly larger improvements than does LCE_HMRF, except on ROBUST; VN-LCE gives an improvement of about 13.42% on WT10G and 11.82% on GOV2; LCE_HMRF gives an improvement of about 8.63% on WT10G and 9.69% on GOV2. This result is remarkable, given that the two-stage normalization is not based on a significant modification of the framework itself, whereas LCE_HMRF results from a sophisticated extension of original LCE. Because two-stage normalization could also be straight forwardly applicable to LCE_HMRF, it would be worthy to investigate the effect of two-stage normalization under the setting of LCE_HMRF and to see whether the additional performance improvement in LCE_HMRF can also be obtained.

In addition, Table 11 shows the performances with NDCG@20 of VN-RM3 and VN-LCE, as compared to RM3 and LCE. The results for NDCG@20 are largely similar to those for the MAP, as seen in Table 9. In many cases, the VN-RM3's and VN-LCE's performances with NDCG20 were better than those of RM3 and LCE, respectively, with statistically significant improvements in most cases. In particular, in GOV2, the improvements of NDCG@20 by VN-PRF (viz., VN-RM3 and VN-LCE) over the original methods (viz., RM3 and LCE) were noticeable, showing a mostly 5–7% increase with NDCG@20.

Overall, the best performance with NDCG@20 was always achieved by either VN-RM3 or VN-LCE, except when $R = 5$ for ROBUST, where LCE showed the best result with NDCG@20. In contrasting VN-RM3 and VN-LCE, the latter was consistently better, except when $R = 30$ with ClueWeb09.

7.3. Comparison of the robustness index: RM3 vs. VN-RM3, and LCE vs. VN-LCE

The commonly recognized characteristic of a PRF method is that there are negatively affected queries in which the initial-retrieval performance does not improve after applying a PRF method. This is because a PRF method relies on top-retrieved documents, and these are an unreliable source of training examples that usually include non-relevant documents as well. This motivated us to evaluate how risky a PRF method is, and to determine how robust a PRF is regardless of the initial-retrieval performance.

To this end, we further computed the robustness index (RI) (Collins-Thompsonand Callan, 2007), which was introduced to check how a PRF method consistently improves the performance of the initial retrieval. Given a set of queries *Q*, RI is defined as follows:

$$RI = \frac{n_+ - n_-}{|Q|} \tag{16}$$

where n_+ and n_- denote the number of queries helped and the number of queries hurt by a PRF method, respectively. Given this definition, RI helps us to check whether the resulting absolute increase in the MAP or NDCG@20 might derive from by making consistent improvements over queries regardless of the performance of the initial retrieval, or whether it is the result of helping a few queries only while actually hurting the majority of queries.

Table 12 presents the overall comparison results of RI between the proposed VN-PRF (i.e., VN-RM3 and VN-LCE) and the original PRF methods (i.e., RM3 and LCE) computed over all test collections, where the total number of queries *N* used for computing RI is presented in the column named "All" in Table 13. As shown the tables, the range of RIs for all PRF methods falls mostly between 0.2 and 0.4. Importantly, the RIs of VN-PRF are larger than those of the original PRF methods, demonstrating that a VN-PRF method is

Table 11
NDCG@20 performance comparison of four pseudo-relevance feedback methods: RM3, VN-RM3, LCE, and VN-LCE. The MAP is used as the evaluation metric. The symbols α, β, and γ indicate statistically significant improvements at the 0.95 confidence level over DP, RM3, and LCE, respectively. The number of expansion terms was 100.

R		ROBUST	WT10G	GOV2	ClueWeb09
Init	DP	0.4207	0.3101	0.4564	0.1869
	MRF	0.4301^{α}	0.3284^{α}	0.4738^{α}	0.1985^{α}
$R = 5$	RM3	0.4401^{α}	0.3190	0.4570	0.1835
	VN-RM3	0.4497^{α}	0.3382^{α}	$0.5113^{\alpha\beta}$	$0.2316^{\alpha\beta}$
	LCE	$\mathbf{0.4582}^{\alpha\beta}$	$0.3413^{\alpha\beta}$	$0.4863^{\alpha\beta}$	$0.2075^{\alpha\beta}$
	VN-LCE	0.4550^{α}	$\mathbf{0.3650}^{\alpha\beta\gamma}$	$\mathbf{0.5461}^{\alpha\beta\gamma}$	$\mathbf{0.2416}^{\alpha\beta\gamma}$
$R = 10$	RM3	0.4337	0.3153	0.4565	0.1895
	VN-RM3	$0.4531^{\alpha\beta}$	0.3350^{α}	$0.5170^{\alpha\beta}$	$0.2326^{\alpha\beta}$
	LCE	$0.4536^{\alpha\beta}$	0.3346^{α}	$0.4850^{\alpha\beta}$	$0.2155^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4619}^{\alpha\beta}$	$\mathbf{0.3608}^{\alpha\beta\gamma}$	$\mathbf{0.5532}^{\alpha\beta\gamma}$	$\mathbf{0.2527}^{\alpha\beta\gamma}$
$R = 15$	RM3	0.4349	0.3148	0.4650	0.1920
	VN-RM3	$0.4505^{\alpha\beta}$	0.3303	$0.5200^{\alpha\beta}$	$0.2371^{\alpha\beta}$
	LCE	$0.4511^{\alpha\beta}$	0.3305^{α}	0.4827^{α}	$0.2108^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4679}^{\alpha\beta\gamma}$	$\mathbf{0.3496}^{\alpha\beta}$	$\mathbf{0.5510}^{\alpha\beta\gamma}$	$\mathbf{0.2574}^{\alpha\beta\gamma}$
$R = 20$	RM3	0.4295	0.3156	0.4623	0.1935
	VN-RM3	$0.4467^{\alpha\beta}$	0.3288	$0.5165^{\alpha\beta}$	$0.2426^{\alpha\beta}$
	LCE	$0.4493^{\alpha\beta}$	0.3272^{α}	$0.4809^{\alpha\beta}$	0.2186
	VN-LCE	$\mathbf{0.4665}^{\alpha\beta\gamma}$	$\mathbf{0.3540}^{\alpha\beta\gamma}$	$\mathbf{0.5499}^{\alpha\beta\gamma}$	$\mathbf{0.2618}^{\alpha\beta\gamma}$
$R = 30$	RM3	0.4276	0.3076	0.4576	0.1960
	VN-RM3	$0.4408^{\alpha\beta}$	0.3250	$0.5172^{\alpha\beta}$	$\mathbf{0.2617}^{\alpha\beta}$
	LCE	$0.4496^{\alpha\beta}$	0.3174	0.4796	$0.2195^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4652}^{\alpha\beta\gamma}$	$\mathbf{0.3513}^{\alpha\beta\gamma}$	$\mathbf{0.5457}^{\alpha\beta\gamma}$	$0.2596^{\alpha\beta\gamma}$
Trained	RM3	0.4371^{α}	0.3172	0.4566	0.1960
	VN-RM3	$0.4492^{\alpha\beta}$	0.3270	$0.5199^{\alpha\beta}$	$\mathbf{0.2617}^{\alpha\beta}$
	LCE	0.4511^{α}	$0.3361^{\alpha\beta}$	0.4818^{α}	$0.2195^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4679}^{\alpha\beta\gamma}$	$\mathbf{0.3650}^{\alpha\beta\gamma}$	$\mathbf{0.5545}^{\alpha\beta\gamma}$	$0.2596^{\alpha\beta\gamma}$

Table 12
Robustness index (RI) comparison of four pseudo-relevance feedback methods, RM3, VN-RM3, LCE, and VN-LCE, for the combined queries from all test collections. As in the setting of Collins-Thompson and Callan (2007), queries for which MAPs of their initial retrieval were very low (≤ 0.01) were ignored. The total number of queries N used to compute RI is presented in the "All" column in Table 13.

R	Method	n_{-}	RI
$R=5$	RM3	225	0.2879
	VN-RM3	200	0.3717
	LCE	231	0.2773
	VN-LCE	206	0.3604
$R=10$	RM3	236	0.2582
	VN-RM3	215	0.3235
	LCE	222	0.3069
	VN-LCE	211	0.3436
$R=15$	RM3	229	0.2817
	VN-RM3	202	0.3608
	LCE	229	0.2835
	VN-LCE	212	0.3436
$R=20$	RM3	228	0.2833
	VN-RM3	208	0.3484
	LCE	243	0.2399
	VN-LCE	209	0.3543
$R=30$	RM3	230	0.2754
	VN-RM	221	0.3079
	LCE	247	0.2274
	VN-LCE	201	0.3788
Trained	RM	222	0.3020
	VN-RM	216	0.3250
	LCE	240	0.2492
	VN-LCE	190	0.4095

Table 13
Total number of queries N for computing robustness index (RI) for RM3, VN-RM3, LCE, and VN-LCE on ROBUST, WT10G, GOV2, and ClueWeb09. Here, "All" refers to the amalgam of all four collections. As in the setting of Collins-Thompson and Callan (2007), queries for which MAPs of their initial retrieval were very low (≤ 0.01) were ignored.

Method	ROBUST	WT10G	GOV2	ClueWeb09	All
RM	241	87	145	166	639
VN-RM	237	88	146	172	643
LCE	238	89	146	169	642
VN-LCE	239	87	148	178	652

more robust than an original PRF method in terms of its dependency on the initial-retrieval performance. The difference in RI between VN-PRF and original PRF methods is largely dependent on R. Specifically, when R is trained (i.e., when it corresponds to the rows named "Trained"), it is remarkable that VN-LCE leads to a more than 0.15 absolute increase of RI over LCE, while VN-RM3 only slightly increases the RI of RM3. When R is 10, VN-RM3 results in a larger increase of RI over RM3 than that of VN-LCE over LCE.

For more detail, Table 14 shows the comparison of the RI between the proposed VN-PRF and the original PRF methods in ROBUST, WT10G, GOV2, and ClueWeb09, separately, where the total number of queries used to compute RIs is presented in Table 13. As shown in the table, whether VN-PRF methods improve RIs of the original PRF methods is highly collection-dependent. Specifically: 1) VN-LCE vs. LCE: the RIs of VN-LCE are higher than LCE in WT10G and ClueWeb09, whereas the RIs of VN-LCE and LCE are similar in ROBUST and GOV2. 2) VN-RM3 vs. RM: RIs of VN-RM3 are mostly higher than those of RM3 in ROBUST, WT10G, and ClueWeb09, whereas the RIs of VN-RM3 are lower than the RIs on GOV2.

Overall, the results in Tables 12 and 14 show that, in general, the proposed VN-PRFs do not increase the number of queries hurt compared to the original PRFs. Rather, they tend to increase the number of queries helped, and often this is a considerable increase, depending on the test collections and the PRF methods used. Therefore, the proposed VN-PRFs are robust, to at least the extent of the degree of the original PRF in the terms of the RI. This confirms that the statistically significant improvements in the MAP and NDCG@20 by the proposed VN-PRFs observed in Tables 9 and 11 are not made from exceptionally large gains restricted to only few queries, but rather from consistent improvements to the number of helped queries and from improvements to at least as many queries as the original PRF method.

Table 14
Robustness index (RI) comparison of four pseudo-relevance feedback methods, RM3, VN-RM3, LCE, and VN-LCE. RI refers to the robustness index in Collins-Thompson and Callan (2007), and n_- is the number of queries hurt. As in the setting of Collins-Thompson and Callan (2007), queries for which MAPs of their initial retrieval were very low (≤ 0.01) were ignored. The row named "Tr." refers to the results from using trained R for each collection.

R	Method	ROBUST		WT10G		GOV2		ClueWeb09	
		n_-	RI	n_-	RI	n_-	RI	n_-	RI
5	RM3	77	0.3610	32	0.2529	52	0.2759	64	0.2108
	VN-RM3	66	0.4430	29	0.3295	39	0.4589	66	0.2209
	LCE	62	0.4790	40	0.0787	55	0.2466	74	0.1243
	VN-LCE	74	0.3766	29	0.2989	37	0.4932	66	0.2584
10	RM3	82	0.3195	39	0.0920	52	0.2828	63	0.2349
	VN-RM3	67	0.4304	29	0.3182	60	0.1781	59	0.3023
	LCE	70	0.4118	39	0.1124	49	0.3288	64	0.2426
	VN-LCE	71	0.4059	32	0.2184	47	0.3649	61	0.3034
15	RM3	86	0.2863	31	0.2759	48	0.3379	64	0.2289
	VN-RM3	67	0.4346	33	0.2273	49	0.3151	53	0.3663
	LCE	79	0.3361	43	0.0112	51	0.3014	56	0.3373
	VN-LCE	68	0.4268	37	0.1379	54	0.2703	53	0.3933
20	RM3	83	0.3112	32	0.2529	49	0.3172	64	0.2289
	VN-RM	73	0.3840	33	0.2273	54	0.2603	48	0.4360
	LCE	83	0.3025	38	0.1236	54	0.2603	68	0.1953
	VN-LCE	69	0.4226	34	0.1954	52	0.2973	54	0.3876
30	RM3	86	0.2863	34	0.2069	46	0.3586	64	0.2229
	VN-RM	73	0.3840	33	0.2159	52	0.2877	63	0.2674
	LCE	80	0.3277	44	-0.0112	54	0.2603	69	0.1834
	VN-LCE	77	0.3556	30	0.2874	51	0.3108	43	0.5112
Tr.	RM	81	0.3278	27	0.3678	50	0.3103	64	0.2229
	VN-RM	69	0.4177	33	0.2386	51	0.2945	63	0.2674
	LCE	79	0.3361	43	0.0112	49	0.3288	69	0.1834
	VN-LCE	68	0.4268	29	0.2989	50	0.3243	43	0.5112

Table 15
MAP performance comparison of four pseudo-relevance feedback methods RM3, VN-RM3, LCE, and VN-LCE. MAP is used as the evaluation metric. The symbols α, β, and γ indicate statistically significant improvements at 0.95 confidence level over DP, RM3, and LCE, respectively. The number of expansion terms is 10.

R		ROBUST	WT10G	GOV2	ClueWeb09
Init	DP	0.2447	0.1963	0.2907	0.1228
	MRF	0.2545^{α}	0.2149^{α}	0.3095^{α}	0.1312^{α}
$R=5$	RM3	0.2683^{α}	0.2171^{α}	0.3030^{α}	0.1213
	VN-RM3	0.2634^{α}	0.2291^{α}	$0.3239^{\alpha\beta}$	$0.1413^{\alpha\beta}$
	LCE	$0.2781^{\alpha\beta}$	0.2200^{α}	$0.3209^{\alpha\beta}$	$0.1341^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2811}^{\alpha\beta}$	$\mathbf{0.2549}^{\alpha\beta\gamma}$	$\mathbf{0.3618}^{\alpha\beta\gamma}$	$\mathbf{0.1516}^{\alpha\beta\gamma}$
$R=10$	RM3	0.2662^{α}	0.2191^{α}	0.3004^{α}	0.1226
	VN-RM3	0.2667^{α}	0.2207^{α}	$0.3209^{\alpha\beta}$	$0.1423^{\alpha\beta}$
	LCE	$0.2888^{\alpha\beta}$	0.2224^{α}	$0.3238^{\alpha\beta}$	$0.1351^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2924}^{\alpha\beta}$	$\mathbf{0.2545}^{\alpha\beta\gamma}$	$\mathbf{0.3649}^{\alpha\beta\gamma}$	$\mathbf{0.1550}^{\alpha\beta\gamma}$
$R=15$	RM3	0.2635^{α}	0.2147^{α}	0.3017^{α}	0.1238
	VN-RM3	0.2625^{α}	0.2211^{α}	$0.3179^{\alpha\beta}$	$0.1428^{\alpha\beta}$
	LCE	$0.2897^{\alpha\beta}$	0.2235^{α}	$0.3256^{\alpha\beta}$	$0.1353^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2981}^{\alpha\beta\gamma}$	$\mathbf{0.2504}^{\alpha\beta\gamma}$	$\mathbf{0.3646}^{\alpha\beta\gamma}$	$\mathbf{0.1569}^{\alpha\beta\gamma}$
$R=20$	RM3	0.2600^{α}	0.2118^{α}	0.2955^{α}	0.1237
	VN-RM3	0.2604^{α}	0.2169^{α}	$0.3163^{\alpha\beta}$	$0.1433^{\alpha\beta}$
	LCE	$0.2878^{\alpha\beta}$	$0.2251^{\alpha\beta}$	$0.3226^{\alpha\beta}$	$0.1356^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2960}^{\alpha\beta\gamma}$	$\mathbf{0.2509}^{\alpha\beta\gamma}$	$\mathbf{0.3632}^{\alpha\beta\gamma}$	$\mathbf{0.1584}^{\alpha\beta\gamma}$
$R=30$	RM3	0.2586^{α}	0.2094^{α}	0.2942	0.1238
	VN-RM3	0.2567^{α}	0.2200^{α}	$0.3164^{\alpha\beta}$	$0.1436^{\alpha\beta}$
	LCE	$0.2861^{\alpha\beta}$	$0.2249^{\alpha\beta}$	$0.3211^{\alpha\beta}$	$0.1363^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2925}^{\alpha\beta\gamma}$	$\mathbf{0.2536}^{\alpha\beta\gamma}$	$\mathbf{0.3599}^{\alpha\beta\gamma}$	$\mathbf{0.1589}^{\alpha\beta\gamma}$
Trained	RM3	0.2659^{α}	0.2176^{α}	0.3007^{α}	0.1210
	VN-RM3	0.2667^{α}	0.2291^{α}	$0.3239^{\alpha\beta}$	$0.1426^{\alpha\beta}$
	LCE	$0.2879^{\alpha\beta}$	0.2158	$0.3203^{\alpha\beta}$	$0.1360^{\alpha\beta}$
	VN-LCE	$\mathbf{0.2981}^{\alpha\beta\gamma}$	$\mathbf{0.2519}^{\alpha\beta\gamma}$	$\mathbf{0.3646}^{\alpha\beta\gamma}$	$\mathbf{0.1586}^{\alpha\beta\gamma}$

7.4. Comparison under a small number of expansion terms: RM3 vs. VN-RM3 and LCE vs. VN-LCE

Until now, we fixed the number of expansion terms as 100 for all experiments. A small number of expansion terms may be preferred frequently in efficiency-aware PRFs because the search time of PRF increases with the number of expansion terms.

In this experiment, we fixed the number of expansion terms as 10 and carried out all experiments described in Section 7.1–7.3 again to examine the change in performance. For notation convenience, we use *full query expansion* (**FQE**) and *small query expansion* (**SQE**) to refer to the cases in which the number of expansion terms is at most 100 and 10, respectively. Similarly, full PRF and restricted PRFs refer to the PRFs that use 100 and 10 expansion terms, respectively.

Tables 15 and 16 compare the performances of VN-RM3 and VN-LCE with RM3 and LCE in terms of MAP and NDCG20, when the number of expansion terms is fixed as 10.

Overall, under SQE, VN-PRF outperforms the original PRF methods and the resulting improvements in terms of MAP and NDCG@20 are statistically significant in most cases. In particular, VN-LCE with SQE exhibits the best performances among all four methods and shows statistically significant improvements over DP, RM3, and LCE in all test collections. In terms of NDCG@20, the improvements provided by VN-PRF (either VN-RM3 or VN-LCE) over original PRF (either RM3 or LCE) are statistically significant under SQE, which is similar to the results for FQE. Specifically:

1) VN-LCE vs. LCE: VN-LCE provides statistically significant improvements over LCE in terms of MAP and NDCG@20. Performance increases significantly, particularly in the WT10G and GOV collections, where increases of more than 4% in MAP and approximately 5–7% in NDCG@20 are obtained. More specifically, in terms of MAP, the differences between the performances of VN-LCE and LCE under SQE are almost similar to those under full PRF. For example, in the run "trained R", the differences between the performances of VN-LCE and LCE in terms of MAP under SQE are 1.02%, 3.61%, 4.43%, and 2.26% on ROBUST, WT10G, GOV2, and ClueWeb09, respectively. These differences are almost similar to 1.23%, 3.08%, 3.95%, and 2.35% which correspond to the cases in which FQE is used on the four collections. In addition, in terms of NDCG@20, the differences between the performances of VN-LCE and LCE under SQE are similar to those under FQE.
2) VN-RM3 vs. RM3: In terms of MAP, the differences between the performances of VN-RM3 and RM3 under SQE are reduced, with an absolute decrease of approximately 0.51%. Nevertheless, improvements of RM3 by VN-RM3 are statistically significant on GOV2 and ClueWeb09. On the contrary, in terms of NDCG@20, the differences between the performances of VN-RM3 and RM3 under SQE are largely similar to those under FQE. The performance improvements provided by VN-RM3 over RM3 are statistically significant for all collections except WT10G.

Table 17 compares the RI computed over all test collections using the VN-PRF and original PRF methods under SQE, where the total number of queries is the same as that in the column labeled "All" in Table 13. As shown in the table, when R is trained, VN-PRFs

Table 16
NDCG@20 performance comparison of four pseudo-relevance feedback methods: RM3, VN-RM3, LCE, and VN-LCE. The MAP is used as the evaluation metric. The symbols α, β, and γ indicate statistically significant improvements at the 0.95 confidence level over DP, RM3, and LCE, respectively. The number of expansion terms was 10.

R		ROBUST	WT10G	GOV2	ClueWeb09
Init	DP	0.4207	0.3101	0.4564	0.1869
	MRF	0.4301^{α}	0.3284^{α}	0.4738^{α}	0.1985^{α}
$R=5$	RM3	0.4285	0.3140	0.4542	0.1796
	VN-RM3	0.4357^{α}	0.3337	$0.4986^{\alpha\beta}$	$0.2211^{\alpha\beta}$
	LCE	0.4385^{α}	0.3255	0.4687	$0.2058^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4482}^{\alpha\beta\gamma}$	$\mathbf{0.3686}^{\alpha\beta\gamma}$	$\mathbf{0.5394}^{\alpha\beta\gamma}$	$\mathbf{0.2326}^{\alpha\beta\gamma}$
$R=10$	RM3	0.4204	0.3253	0.4485	0.1820
	VN-RM3	$0.4352^{\alpha\beta}$	0.3234	$0.5077^{\alpha\beta}$	$0.2202^{\alpha\beta}$
	LCE	$0.4405^{\alpha\beta}$	0.3290	0.4703	$0.2045^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4529}^{\alpha\beta\gamma}$	$\mathbf{0.3653}^{\alpha\beta\gamma}$	$\mathbf{0.5395}^{\alpha\beta\gamma}$	$\mathbf{0.2396}^{\alpha\beta\gamma}$
$R=15$	RM3	0.4209	0.3207	0.4523	0.1847
	VN-RM3	0.4296^{α}	0.3238	$0.5051^{\alpha\beta}$	$0.2253^{\alpha\beta}$
	LCE	$0.4420^{\alpha\beta}$	0.3261	0.4738	$0.2074^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4571}^{\alpha\beta\gamma}$	$\mathbf{0.3587}^{\alpha\beta\gamma}$	$\mathbf{0.5379}^{\alpha\beta\gamma}$	$\mathbf{0.2470}^{\alpha\beta\gamma}$
$R=20$	RM3	0.4152	0.3170	0.4481	0.1846
	VN-RM3	0.4251	0.3254	$0.5037^{\alpha\beta}$	$0.2266^{\alpha\beta}$
	LCE	$0.4371^{\alpha\beta}$	0.3315	0.4663	$0.2074^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4552}^{\alpha\beta\gamma}$	$\mathbf{0.3614}^{\alpha\beta\gamma}$	$\mathbf{0.5368}^{\alpha\beta\gamma}$	$\mathbf{0.2526}^{\alpha\beta\gamma}$
$R=30$	RM3	0.4112	0.3112	0.4432	0.1856
	VN-RM3	0.4232^{β}	0.3300	$0.5035^{\alpha\beta}$	$0.2273^{\alpha\beta}$
	LCE	$0.4353^{\alpha\beta}$	0.3304^{β}	0.4616	$0.2103^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4517}^{\alpha\beta\gamma}$	$\mathbf{0.3627}^{\alpha\beta\gamma}$	$\mathbf{0.5344}^{\alpha\beta\gamma}$	$\mathbf{0.2545}^{\alpha\beta\gamma}$
Trained	RM3	0.4236	0.3224	0.4499	0.1794
	VN-RM3	$0.4352^{\alpha\beta}$	0.3337	$0.4986^{\alpha\beta}$	$0.2259^{\alpha\beta}$
	LCE	$0.4389^{\alpha\beta}$	0.3179	0.4722^{β}	$0.2100^{\alpha\beta}$
	VN-LCE	$\mathbf{0.4571}^{\alpha\beta\gamma}$	$\mathbf{0.3667}^{\alpha\beta\gamma}$	$\mathbf{0.5404}^{\alpha\beta\gamma}$	$\mathbf{0.2520}^{\alpha\beta\gamma}$

Table 17
Robustness index (RI) comparison of four pseudo-relevance feedback methods, RM3, VN-RM3, LCE, and VN-LCE with the number of expansion terms at 10, for the combined queries from all test collections. As in the setting of Collins-Thompson and Callan (2007), queries for which MAPs of their initial retrieval were very low (≤ 0.01) were ignored. The total number of queries N used to compute RI is presented in the "All" column in Table 13.

R	Method	n_-	RI
$R=5$	RM3	247	0.2191
	VN-RM3	259	0.1882
	LCE	251	0.2150
	VN-LCE	243	0.2454
$R=10$	RM3	265	0.1612
	VN-RM3	255	0.1975
	LCE	257	0.1978
	VN-LCE	222	0.3083
$R=15$	RM3	270	0.1487
	VN-RM3	266	0.1695
	LCE	263	0.1729
	VN-LCE	225	0.3021
$R=20$	RM3	262	0.1721
	VN-RM3	266	0.1649
	LCE	266	0.1667
	VN-LCE	221	0.3160
$R=30$	RM3	269	0.1487
	VN-RM3	255	0.1991
	LCE	262	0.1776
	VN-LCE	226	0.3021
Trained	RM3	259	0.1847
	VN-RM3	250	0.2115
	LCE	267	0.1636
	VN-LCE	224	0.3052

Table 18
Robustness index (RI) comparison of four pseudo-relevance feedback methods, RM3, VN-RM3, LCE, and VN-LCE with the number of expansion terms at 10. RI refers to the robustness index in Collins-Thompson and Callan (2007), and n_- is the number of queries hurt. As in the setting of Collins-Thompson and Callan (2007), queries for which MAPs of their initial retrieval were very low (≤ 0.01) were ignored. The row named "Tr." refers to the results from using trained R for each collection.

R	Method	ROBUST		WT10G		GOV2		ClueWeb09	
		n_-	RI	n_-	RI	n_-	RI	n_-	RI
5	RM3	84	0.3029	41	0.0460	53	0.2690	69	0.1446
	VN-RM3	88	0.2532	41	0.0568	57	0.2192	73	0.1395
	LCE	83	0.3025	38	0.1348	60	0.1781	70	0.1657
	VN-LCE	90	0.2427	27	0.3448	55	0.2568	71	0.1910
10	RM3	103	0.1452	33	0.2069	53	0.2690	76	0.0663
	VN-RM3	79	0.3249	38	0.1136	59	0.1918	79	0.0698
	LCE	91	0.2353	42	0.0449	56	0.2329	68	0.1953
	VN-LCE	77	0.3431	27	0.3448	52	0.2973	66	0.2528
15	RM3	103	0.1452	35	0.1839	55	0.2345	77	0.0602
	VN-RM3	83	0.2954	44	-0.0114	62	0.1507	77	0.1047
	LCE	86	0.2773	46	-0.0562	51	0.2877	80	0.0473
	VN-LCE	82	0.3096	30	0.2759	52	0.2973	61	0.3090
20	RM3	96	0.2033	36	0.1609	54	0.2483	76	0.0663
	VN-RM3	86	0.2658	45	-0.0341	64	0.1233	71	0.1628
	LCE	87	0.2647	44	0.0000	55	0.2397	80	0.0533
	VN-LCE	79	0.3347	29	0.3103	53	0.2838	60	0.3202
30	RM3	99	0.1743	39	0.0920	57	0.2069	74	0.0904
	VN-RM3	85	0.2785	39	0.1023	57	0.2123	74	0.1279
	LCE	88	0.2563	42	0.0337	55	0.2397	77	0.0888
	VN-LCE	83	0.3013	29	0.3103	55	0.2568	59	0.3371
Tr.	RM3	94	0.2199	34	0.2069	57	0.2069	74	0.1024
	VN-RM3	79	0.3249	41	0.0568	57	0.2192	73	0.1279
	LCE	89	0.2521	47	-0.0674	56	0.2192	75	0.1124
	VN-LCE	82	0.3096	27	0.3448	55	0.2568	60	0.3202

have higher RIs than original PRFs; this is considerably similar to the results of FQE. Similar to FQE, it is remarkable that VN-LCE leads to an absolute increase of approximately 0.14 in RI, as compared to LCE. However, VN-RM3 only slightly increases the RI of RM3 and sometimes decreases RIs depending on the selection of R.

Table 18 shows the detailed comparison of the RI obtained using the proposed VN-PRF and original PRF methods under SQE for all four test collections. The overall results obtained under SQE are considerably similar to those obtained under FQE, except that the RIs of VN-RM3 are weak on WT10G. In the case of SQE, the absolute values of RI decrease for the VN-PRF and original PRF methods because the improvements provided by the PRF methods become slightly smaller under SQE, as compared to FQE. Interestingly, on WT10G, the RIs of VN-LCE under SQE (0.2759–0.3448) are higher than those under FQE (0.1379–0.2989).

In summary, the overall results confirm that the VN-PRF methods are more effective and robust than the PRF methods even under SQE, particularly for VN-LCE, even though the performance differences between VN-PRF and original PRF in terms of MAP decreases slightly on ROBUST and WT10G in some cases. In addition, the RIs of the VN-PRF methods are higher than those of the original PRF methods in most cases; this is similar to FQE cases.

Consequently, the overall results in Tables 9–18 provide evidence that the verbosity-preserving and verbosity-sensitive effects on feedback query models must be considered in PRF. The experimental results also strengthen the previous effects of two-stage normalization observed in (Na, 2015). The improvements resulting from two-stage normalization are not replaceable by the query-expansion effect of PRF, and the effect of normalization is not weakened, even against a more improved baseline performance.

8. Conclusion

Given the set of results described throughout this paper, we show the following. The proposed two-stage normalization method is clearly effective for further improving the existing retrieval model. It is not limited in its application to proving the baseline retrieval model; moreover, it can be extended, even to the PRF under the LCE framework. Experimental results support our motivating arguments that a PRF method needs to be designed by considering the verbosity-preserving and verbosity-sensitive effects on feedback query models. Therefore, a significant direction in future work would be to extend new retrieval constraints by considering the verbosity-preserving and verbosity-sensitive effects in PRF, in addition to existing constraints for PRF (Clinchant & Gaussier, 2011, 2013; Hazimeh & Zhai, 2015).

An interesting future issue is development of a method that employs other advanced LCE-based methods, including LCE_HMRF (Lang et al., 2010), PQE (Bendersky et al., 2011b,a), and multiple source-based expansion (Bendersky et al., 2012), and to determine how the results can be further improved. It would be additionally interesting to explore the two-stage document length normalization under the setting of cluster-based retrievals, passage-based retrievals, or the query-verbosity-aware retrieval models of (Gupta & Bendersky, 2015; Cummins, 2016). As mentioned in Cummins (2016), it is worthwhile to identify a principled way of incorporating perplexity as a scope measure under a well-founded statistical generative process, such as in the SPUD model.

Acknowledgments

This research was supported by “Research Base Construction Fund Support Program” funded by Chonbuk National University in 2016.

References

Abdul-jaleel, N., Allan, J., Croft, W. B., Diaz, O., Larkey, L., Li, X., et al. (2004). *Umass at TREC 2004: Novelty and hard. Proceedings of the thirteenth text REtrieval conferenceTREC ’04*.

Allan, J., Connell, M. E., Croft, W. B., Feng, F., Fisher, D., & Li, X. (2000). *INQUERY and TREC-9. Proceedings of the ninth text retrieval conference (trec-9)TREC-9*.

Amati, G., & Rijsbergen, C. J. V. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS), 20*, 357–389.

Bendersky, M., Croft, W. B., & Diao, Y. (Croft, Diao, 2011a). *Quality-biased ranking of web documents. Proceedings of the fourth acm international conference on web search and data miningWSDM ’11*95–104.

Bendersky, M., Metzler, D., & Croft, W. B. (2010). *Learning concept importance using a weighted dependence model. Proceedings of the third acm international conference on web search and data miningWSDM ’10*31–40.

Bendersky, M., Metzler, D., & Croft, W. B. (Metzler, Croft, 2011b). *Parameterized concept weighting in verbose queries. Proceedings of the 34th annual international acm sigir conference on research and development in information retrievalSIGIR ’11*.

Bendersky, M., Metzler, D., & Croft, W. B. (2012). *Effective query formulation with multiple information sources. Proceedings of the fifth acm international conference on web search and data miningWSDM ’12*443–452.

Chung, T. L., Luk, R. W. P., Wong, K. F., Kwok, K. L., & Lee, D. L. (2006). Adapting pivoted document-length normalization for query size: Experiments in chinese and english. *ACM Transactions on Asian Language Information Processing (TALIP), 5*(3), 245–263.

Clinchant, S., & Gaussier, É. (2011). *A document frequency constraint for pseudo-relevance feedback models. Conférence en recherche d’infomations et applications - 8th french information retrieval conferenceCORIA ’11*73–88.

Clinchant, S., & Gaussier, E. (2013). *A theoretical analysis of pseudo-relevance feedback models. Proceedings of the 2013 conference on the theory of information retrievalICTIR ’13*6:6–6:13.

Collins-Thompson, K., & Callan, J. (2007). *Estimation and use of uncertainty in pseudo-relevance feedback. Proceedings of the 30th annual international acm sigir conference on research and development in information retrievalSIGIR ’07*303–310.

Croft, W. B., Metzler, D., & Strohman, T. (2009). *Search engines: information retrieval in practice* (1st). Addison-Wesley Publishing Company.

Cummins, R. (2016). *A study of retrieval models for long documents and queries in information retrieval. Proceedings of the 25th international world wide web conferenceWWW’16*.

Cummins, R., & O’Riordan, C. (2012). *A constraint to automatically regulate document-length normalisation. Proceedings of the 21st acm international conference on information and knowledge managementCIKM ’12*2443–2446.

Cummins, R., Paik, J. H., & Lv, Y. (2015). A pólya urn document language model for improved information retrieval. *ACM Transactions on Information Systems, 33*(4), 21:1–21:34.

Di Buccio, E., Melucci, M., & Moro, F. (2014). Detecting verbose queries and improving information retrieval. *Information Processing and Management, 50*(2), 342–360.

Fang, H., Tao, T., & Zhai, C. (2004). *A formal study of information retrieval heuristics. Proceedings of the 27th annual international acm sigir conference on research and development in information retrievalSIGIR ’04*49–56.

Fang, H., Tao, T., & Zhai, C. (2011). Diagnostic evaluation of information retrieval models. *ACM Transactions on Information Systems (TOIS), 29*, 7:1–7:42.

Gupta, M., & Bendersky, M. (2015). *Information retrieval with verbose queries. Proceedings of the 38th international acm sigir conference on research and development in information retrievalSIGIR ’15*1121–1124.

Hazimeh, H., & Zhai, C. (2015). *Axiomatic analysis of smoothing methods in language models for pseudo-relevance feedback. Proceedings of the 2015 international conference on the theory of information retrievalICTIR ’15*141–150.

He, B., & Ounis, I. (2003). *A study of parameter tuning for term frequency normalization. Proceedings of the twelfth international conference on information and knowledge managementCIKM ’03*10–16.

Hui, K., He, B., Luo, T., & Wang, B. (2011). *A comparative study of pseudo relevance feedback for ad-hoc retrieval. Proceedings of the 2011 conference on the theory of information retrievalICTIR ’11*318–322.

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems, 20*(4), 422–446.

Karisani, P., Rahgozar, M., & Oroumchian, F. (2016). A query term re-weighting approach using document similarity. *Information Processing and Management, 52*(3), 478–489.

Kurland, O., & Lee, L. (2005). *Pagerank without hyperlinks: structural re-ranking using links induced by language models. Proceedings of the 28th annual international acm sigir conference on research and development in information retrievalSIGIR ’05*306–313.

Lafferty, J., & Zhai, C. (2001). *Document language models, query models, and risk minimization for information retrieval. Proceedings of the 24th annual international acm sigir conference on research and development in information retrievalSIGIR ’01*111–119.

Lang, H., Metzler, D., Wang, B., & Li, J.-T. (2010). *Improved latent concept expansion using hierarchical markov random fields. Proceedings of the 19th acm international conference on information and knowledge managementCIKM ’10*249–258.

Lavrenko, V. (2004). *A generative theory of relevance*. University of Massachusetts Amherst Ph.D. thesis.

Lavrenko, V., & Croft, W. B. (2001). *Relevance based language models. Proceedings of the 24th annual international acm sigir conference on research and development in information retrievalSIGIR ’01*120–127.

Lipani, A., Lupu, M., Hanbury, A., & Aizawa, A. (2015). *Verboseness Fission for BM25 Document Length Normalization. Proceedings of the 2015 international conference on the theory of information retrievalICTIR ’15*385–388.

Lv, Y. (2015). *A study of query length heuristics in information retrieval. Proceedings of the 24th acm international on conference on information and knowledge managementCIKM ’15*1747–1750.

Lv, Y., & Zhai, C. (Zhai, 2009a). *A comparative study of methods for estimating query language models with pseudo feedback. Proceeding of the 18th acm conference on information and knowledge managementCIKM ’09*1895–1898.

Lv, Y., & Zhai, C. (Zhai, 2009b). *Positional language models for information retrieval. Proceedings of the 32nd international acm sigir conference on research and development in information retrievalSIGIR ’09*299–306.

Lv, Y., & Zhai, C. (2010). *Positional relevance model for pseudo-relevance feedback. Proceeding of the 33rd international acm sigir conference on research and development in information retrievalSIGIR ’10*579–586.

Lv, Y., & Zhai, C. (Zhai, 2011a). *Adaptive term frequency normalization for BM25. Proceedings of the 20th acm international conference on information and knowledge managementCIKM ’11*1985–1988.

Lv, Y., & Zhai, C. (Zhai, 2011b). *Lower-bounding term frequency normalization. Proceedings of the 20th acm international conference on information and knowledge managementCIKM ’11*7–16.

Lv, Y., & Zhai, C. (Zhai, 2011c). *When documents are very long, BM25 fails!. Proceedings of the 34th international acm sigir conference on research and development in information retrievalSIGIR ’11*1103–1104.

Metzler, D., & Croft, W. B. (2005). *A markov random field model for term dependencies. Proceedings of the 28th annual international acm sigir conference on research and development in information retrievalSIGIR ’05*472–479.

Metzler, D., & Croft, W. B. (2007). *Latent concept expansion using markov random fields. Proceedings of the 30th annual international acm sigir conference on research and development in information retrievalSIGIR '07*311–318.
Na, S.-H. (2015). Two-stage document length normalization for information retrieval. *ACM Transactions on Information Systems (TOIS), 33*(2), 8:1–8:40.
Na, S.-H., Kang, I.-S., & Lee, J.-H. (2008). *Improving term frequency normalization for multi-topical documents and application to language modeling approaches. Proceedings of the ir research, 30th european conference on advances in information retrievalECIR'08*382–393.
Paik, J. H. (2013). *A Novel TF-IDF Weighting Scheme for Effective Ranking. Proceedings of the 36th international acm sigir conference on research and development in information retrievalSIGIR '13*343–352.
Paik, J. H. (2015). *A probabilistic model for information retrieval based on maximum value distribution. Proceedings of the 38th international acm sigir conference on research and development in information retrievalSIGIR '15*585–594.
Robertson, S. E., & Walker, S. (1994). *Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. Proceedings of the 17th annual international acm sigir conference on research and development in information retrievalSIGIR '94*232–241.
Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., & Gatford, M. (1995). *Okapi at TREC-3. Proceedings of the thrid text retrieval conference (trec-3)TREC-3*.
Robertson, S. E., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval, 3*(4), 333–389.
Rousseau, F., & Vazirgiannis, M. (Vazirgiannis, 2013a). *Composition of TF Normalizations: New Insights on Scoring Functions for Ad Hoc IR. Proceedings of the 36th international acm sigir conference on research and development in information retrievalSIGIR '13*917–920.
Rousseau, F., & Vazirgiannis, M. (Vazirgiannis, 2013b). *Graph-of-word and TW-IDF: New Approach to Ad Hoc IR. Proceedings of the 22nd acm international conference on information & knowledge managementCIKM '13*59–68.
Singhal, A., Buckley, C., & Mitra, M. (1996). *Pivoted document length normalization. Proceedings of the 19th annual international acm sigir conference on research and development in information retrievalSIGIR '96*21–29.
Smucker, M. D., & Allan, J. (2005). *An investigation of dirichlet prior smoothingś performance advantageTechnical Report.* CIIR Technical Report IR-548 (University of Massachusetts, Amherst).
Ye, Z., He, B., Huang, X., & Lin, H. (2010). *Revisiting rocchio's relevance feedback algorithm for probabilistic models. Proceedings of the 6th asia information retrieval societies conferenceAIRS '10*151–161.
Zhai, C., & Lafferty, J. (Lafferty, 2001a). *Model-based feedback in the language modeling approach to information retrieval. Proceedings of the tenth international conference on information and knowledge managementCIKM '01*403–410.
Zhai, C., & Lafferty, J. (Lafferty, 2001b). *A study of smoothing methods for language models applied to ad hoc information retrieval. Proceedings of the 24th annual international acm sigir conference on research and development in information retrievalSIGIR '01*334–342.
Zhai, C., & Lafferty, J. (2006). A risk minimization framework for information retrieval. *Information Processing and Management, 42*, 31–55.
Zhang, P., Song, D., Wang, J., & Hou, Y. (2014). Bias-variance analysis in estimating true query model for information retrieval. *Information Processing and Management, 50*(1), 199–217.