

A Vertical PRF Architecture for Microblog Search

Flávio Martins
NOVA LINCS
Faculty of Science and Technology
Universidade NOVA de Lisboa
Caparica, Portugal
flaviomartins@acm.org

João Magalhães
NOVA LINCS
Faculty of Science and Technology
Universidade NOVA de Lisboa
Caparica, Portugal
jm.magalhaes@fct.unl.pt

Jamie Callan
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, USA
callan@cs.cmu.edu

ABSTRACT

In microblog retrieval, query expansion can be essential to obtain good search results due to the short size of queries and posts. Since information in microblogs is highly dynamic, an up-to-date index coupled with pseudo-relevance feedback (PRF) with an external corpus has a higher chance of retrieving more relevant documents and improving ranking. In this paper, we focus on the research question: *how can we reduce the query expansion computational cost while maintaining the same retrieval precision as standard PRF?* Therefore, we propose to accelerate the query expansion step of pseudo-relevance feedback. The hypothesis is that using an expansion corpus organized into verticals for expanding the query, will lead to a more efficient query expansion process and improved retrieval effectiveness. Thus, the proposed query expansion method uses a distributed search architecture and resource selection algorithms to provide an efficient query expansion process. Experiments on the TREC Microblog datasets show that the proposed approach can match or outperform standard PRF in MAP and NDCG@30, with a computational cost that is three orders of magnitude lower.

ACM Reference Format:

Flávio Martins, João Magalhães, and Jamie Callan. 2018. A Vertical PRF Architecture for Microblog Search. In *ICTIR '18: 2018 ACM SIGIR International Conference on the Theory of Information Retrieval*, Sept. 14–17, 2018, Tianjin, China. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3234944.3234960>

1 INTRODUCTION

In microblogs there is a high mismatch between the keywords users employ to specify the information need and the words in relevant documents, which is known as the *vocabulary mismatch problem*. Query expansion is often used to increase recall, however it can also be used to produce better document rankings and increase retrieval effectiveness. Therefore, query expansion methods based on PRF have been widely used to improve search in different collections. It was shown to be an essential feature for microblog search [17].

In standard PRF the top- k documents returned by the initial query (feedback documents) are assumed to be relevant, which avoids the need for users' relevance feedback. Term weights can

be calculated using collection statistics, such as in the model-based relevance models (RM) approach [16]. Several automatic query expansion methods leveraged on external static data such as dictionaries, domain-specific thesauri or precomputed corpus-specific information. In microblogs, query expansion should be based on information that has a good coverage of real-world events.

In order to compute the expansion terms for a query using PRF, it is necessary to issue an extra initial retrieval over the whole collection, which significantly increases the computational complexity of a query. In most production retrieval systems, caching of search results and caching of posting lists are extensively used to alleviate efficiency concerns since it significantly reduces the workload of back-end servers, especially for popular queries and query terms, and provides shorter average response times [3]. However, even in static collections, i) query expansion is done on queries, not query terms; and ii) 20% of all unique queries have not been seen "before"¹, thus, only a few queries can be cached [30]. This problem may be worse in dynamic collections. Therefore, the ephemeral nature of information seeking in microblog search, calls for an architecture that provides fresh expansion terms for better retrieval results.

This paper focuses on the research question *how can we reduce the query expansion computational cost while maintaining the same retrieval precision as standard PRF?* The proposed solution brings two major advantages over standard query expansion approaches in microblogs. First, we reformulate the query expansion process as a federated query expansion task [25], and leverage on resource selection algorithms to select *query-specific* information verticals. Following the *federated search* [7] terminology, a *resource* can be a *source* or a *vertical*. We define a *vertical* (e.g., politics, technology, sports) to be a *query-likelihood* search engine running on a corpus formed by the union of samples from that vertical's corresponding *sources* (e.g., politico, wired, espn). The novel vertical feedback design choice is crucial to unlock the efficiency potential of the proposed query expansion architecture. Second, we depart from previous work that mainly use one single static information corpus and move towards an architecture where *multiple information streams are constantly feeding the query expansion corpus*. This rationale is a step change in the way query expansion is approached: the expansion corpus is constantly updated with fresh data, it is external to the main index, and is segmented into predefined broad topics of interest. To our knowledge, there is no prior work approaching *query expansion with an external and dynamic vertical corpora*. This paradigm shift allows a significant reduction in the document expansion data that is now limited to a few verticals of news sources and therefore allows a faster query expansion process.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICTIR'18, September 14–17, 2018, Tianjin, China

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5656-5/18/09...\$15.00

<https://doi.org/10.1145/3234944.3234960>

¹<https://europe.googleblog.com/2010/02/this-stuff-is-tough.html>

The Pseudo-Relevant Vertical Feedback (PRVF), discussed in section 3, reduces the work-load of the whole search engine for query expansion because it selects a few news verticals and only those are then searched to retrieve feedback documents for query expansion. Coupled with effective resource selection algorithms it outperformed standard PRF in our experiments in section 4 and 5.

2 RELATED WORK

Most microblog search queries could be classified as informational: users issue a query because they are looking for more information about a subject but cannot easily clarify their query intent. Queries submitted to Twitter were found to be significantly shorter than queries submitted to Web search engines (1.64 words vs. 3.08 words) [27], therefore, query expansion is essential to provide a richer description of the information need. The use of named entities and *hashtags* in queries inspired methods that learn a feedback language model for entities [11] and *hashtags* [10]. Additionally, there have been a few pseudo-relevance feedback methods proposed for microblog search that exploit temporal evidences [6, 19, 31].

Pseudo-relevance feedback (PRF) is an automatic query expansion technique, which was shown to significantly improve results in microblog retrieval [17]. It is a popular technique that has been applied in TREC evaluation campaigns on diverse corpora with great effectiveness. However, it is not yet a standard feature in most production search engines, which require faster response times, because it raises efficiency issues at query time.

The standard implementation of PRF involve a two-retrieval process 1) an initial retrieval using the original query to get feedback documents and generate the expanded query, and 2) a re-retrieval using the final expanded query. A recently proposed method, Condensed List Relevance Models (CLRM) [8], foregoes this expensive re-retrieval step by replacing it with the re-ranking of the original feedback documents, which can produce near identical effectiveness in traditional TREC corpora where documents are usually longer. Previous research improves the efficiency of the whole PRF process using a precomputed document similarity matrix [5, 15]. However, these techniques might not be feasible in dynamic collections because term associations would need to be updated constantly.

Several studies have shown enhanced retrieval performance when leveraging on an external corpus. Diaz and Metzler [9] estimate relevance models using an auxiliary large external corpus instead of the target collection and have shown improved retrieval effectiveness on several datasets. In some cases, to improve the effectiveness of query expansion, a large external corpus that is reliable and possibly from less noisy data sources is used. For instance, Arguello et al. [2] used Wikipedia, on a blog retrieval system, and showed significant overall improvements in effectiveness over using the target blog collection for feedback. Xu et al. [32] expanded on this and proposed an approach that categorizes queries based on reliable information from Wikipedia to perform a query-dependent query expansion process based on the category detected.

Bendersky et al. [4] argue that employing multiple information sources in a number of information retrieval processes is desirable to enhanced retrieval, including query expansion. Previous research explored the effectiveness of query expansion in federated

search [21, 24]. First, Ogilvie and Callan [21] analyzed the effectiveness of query expansion in a “one query fits all” fashion, the top-ranked documents retrieved from an index with a representative sample of all the collections are used as feedback to extract terms and create an expanded query that is then submitted to each of the selected search engines. Later, Shokouhi et al. [24] proposed a method that uses a *focused* expanded query for each selected *source* (query-specialization). These provided evidence of the feasibility of query expansion in a federated search environment effectively. We depart from previous work to develop a scalable query expansion index architecture for PRF. In this paper the query expansion index is external to the search collection and is partitioned into verticals to improve efficiency and effectiveness. The architecture proposed can handle a large number of sources or topics, which can be expanded to provide a broader subject coverage.

3 FEDERATED QUERY EXPANSION

We propose a novel solution that provides a balance between effectiveness and efficiency arising from a) the organization of the expansion corpus into verticals, and b) the best performing resource selection algorithms. There are two main types of resource selection algorithms: 1) sample-document methods and 2) vocabulary-based methods. In sample-document methods, a *centralized sample index* (CSI) is built with a representation set for each source or vertical. Representation sets are usually a small sample of documents of about 1-10%, which makes the resource selection process more efficient due to the small size of the resulting CSI index. Two of the most successful sample-document methods Rank-S [14] and *Central-rank-based collection selection* (CRCS) [23] use a similar strategy to the earlier ReDDE [26] algorithm. The user’s query is run against the CSI and the top- n retrieved documents are used in the algorithm in a voting fashion. Vocabulary-based resource selection algorithms, such as Taily [1], represent each collection by its vocabulary statistics only. Previous studies have shown that these approaches are highly effective in reducing the number of search engines that need to be searched.

In PRF the computational cost of the query expansion process is tied to the cost of the initial retrieval. Tackling this major challenge, requires efficient query expansion architectures that reduce this retrieval cost and can still deliver high-quality query expansions.

3.1 The Computational Cost of PRF

The standard PRF procedure creates an expanded query with new terms extracted from the top- k documents in the initial ranking obtained by retrieving with the original query terms. Firstly, the user’s query q^\varnothing is issued to the system to retrieve a ranked list of documents $R(q^\varnothing, D)$, over the whole collection D using a *query-likelihood* (QL) retrieval model. Secondly, the top- k documents retrieved, $R_k(q^\varnothing, D)$, are used to build an expansion language model q^e with the terms extracted from those documents. Finally, the final ranking is obtained by issuing the final query q , which is a linear model combination of the original query language model q^\varnothing and the expansion language model q^e with parameter λ , as follows:

$$q = (1 - \lambda)q^\varnothing + \lambda q^e. \quad (1)$$

The PRF cost can be expressed as the sum of two components: $C_{QE}(q^\varnothing, D)$, the cost of retrieving the ranked list of pseudo-relevant documents using the original query q^\varnothing , and $C_R(q, D)$, the cost of retrieving the final documents using the final expanded query q . Formally, the PRF cost is defined as

$$C_{PRF}(q^\varnothing, D) = C_{QE}(q^\varnothing, D) + C_R(q, D), \quad (2)$$

where the cost metric C_R is based on previous work showing that the sum of the lengths of the posting lists that need to be accessed for each query is strongly correlated with query response times [18, 20]. Hence, for standard PRF, $C_{QE}(q^\varnothing, D) = C_R(q^\varnothing, D)$, and following [1, 13, 14, 18, 20], we define C_R as follows.

Definition 3.1 (Single-step retrieval cost). The cost of retrieval for a given query q is calculated as

$$C_R(q^\varnothing, D) = \sum_{t \in \bar{q}} \text{postings}(t), \quad (3)$$

where $\text{postings}(t)$ is the number of accessed postings in the inverted index for term t .

In the relevance model approach to PRF, the term selection step is constant for all PRF methods when we consider a fixed top-k number of documents. Hence, we discard this part of the cost because we are interested in relative costs.

3.2 Expansion with External Corpus

The use of an external corpus in query expansion has been studied in fields as far apart as blog search and Web search [2, 9, 29]. Arguello et al. [2] addressed blog search with Wikipedia as an alternative query expansion corpus with significant improvements. Freebase has been used for feedback to offer a wide coverage for past events and entities [11]. In light of our goal, we aim to use *the most up-to-date, reliable, and concise external corpus*.

To create a microblog expansion corpus there are several possible strategies. Sampling posts from Twitter accounts picked at random can lead to a low quality expansion collection. Instead, using multiple authoritative news sources lends redundancy to the system since the same news story is often reported by multiple news sources. Hence, since many news outlets use Twitter for the dissemination of news articles, we propose to listen the stream of news headlines directly from their Twitter profile pages (i.e., *timelines*). This corpus can be several orders of magnitude smaller than the target retrieval corpus.

The implemented approach relies on an external news corpus covering multiple authoritative sources for expanding the query (we used 70 news sources). The news corpus is highly dynamic and is maintained up-to-date as new documents are arriving to be indexed – current events are reported live as they unfold by online news sources. As a consequence of this dynamic environment, the *query expansion corpus age and time span* will play a major role in the quality of the expansion corpus.

The use of news sampled from Twitter covers the information seeking behavior of users in the microblog search scenario. This is nicely tied to the natural topical bias of each query, suggesting that partitioning the expansion corpus into news verticals will bring greater benefits in terms of precision and expansion cost.

3.3 PRVF: Pseudo-Relevant Vertical Feedback

The federated query expansion architecture, stems from a new understanding of how temporal and topical information is searched in microblogs. To take full advantage of the external expansion corpus the organization of documents into index shards is fundamental. This architectural decision influences the latency and efficiency of the query expansion process. Uniform sharding distributes the work across all machines so that it can be done more quickly, but it does not reduce the total work done and all the shards are involved for every search query. Previous research found that topic-based shards offer the best balance of retrieval effectiveness and efficiency (query processing computation costs) [12, 14]. Hence, we organize news sources into topic-based verticals, see Table 1.

Queries are routed through a *broker* to a subset of the most useful verticals. To make this decision, the broker keeps a *central sample index* (CSI) of all verticals to select the few verticals to search for each query. This reduces the amount of work done for each query and since the selected verticals can be searched in parallel this approach can be much faster. In a *cooperative* [25] federated search environment, global corpus statistics can be accessed by each federated search engine and by the *broker* and therefore merging results from multiple verticals is straightforward.

Pseudo-Relevant Vertical Feedback (PRVF) is a query expansion architecture that uses an external corpus organized into verticals to efficiently select expansion terms. In the proposed approach, the query expansion corpus is organized into a set of verticals $D_V = \{D_{V1}D_{V2} \cdots D_{V|D_V|}\}$ from which a resource selection method selects the most likely set $\text{sel}(q^\varnothing)$, which are then searched in parallel. Formally, we wish to compute

$$\text{sel}(q^\varnothing) = \{D_{V_{m(1)}}D_{V_{m(2)}} \cdots D_{V_{m(|\text{sel}(q^\varnothing)|)}}\}, \quad (4)$$

where $m : \{1 \cdots |\text{sel}(q^\varnothing)|\} \rightarrow \{1 \cdots |D_V|\}$ is a mapping function that indicates the set of $|\text{sel}(q^\varnothing)|$ verticals selected given q^\varnothing , which are the most promising, in terms of relevance, from the full set of $|D_V|$ verticals.

To select the verticals, a resource selection algorithm either (i) uses a *centralized sample index* (CSI) which indexes a representation sample D_{CSI} [14, 23] of each vertical's documents, or (ii) uses the term statistics [1] of each vertical index. With CSI based algorithms, we retrieve $R_k(q^\varnothing, D_{CSI})$, the top- k documents from collection D_{CSI} in response to the initial query q^\varnothing , using the *query-likelihood* retrieval model. The verticals with more results in this sample are then selected for the feedback retrieval step. The key details of the implemented resource selection algorithms CRCS [23], Rank-S [14] and Taily [1] are in Section 2.

Finally, the top- k documents retrieved from the verticals selected $\text{sel}(q^\varnothing)$ in response to q^\varnothing are merged and used for feedback, *vertical feedback*, to build the expansion language model q^e ,

$$R_k(q^\varnothing, D_V) = \bigcup_{i=1}^{|\text{sel}(q^\varnothing)|} R_k(q^\varnothing, D_{V_{m(i)}}) \quad (5)$$

which is interpolated with the original query model q^\varnothing . This set of documents is then used to expand the original query, thus, ending the computation of q .

3.3.1 Computational cost of PRVF. It is worth recalling equation 2, where we defined the cost of standard PRF. Now, with Pseudo-Relevant Vertical Feedback (PRVF), the query expansion cost C_{QE} is associated to C_{VF} , the cost of performing *vertical feedback*. Formally, the PRVF cost is defined as

$$C_{PRVF}(q) = C_{VF}(q^{\otimes}, D_V) + C_R(q, D), \quad (6)$$

where C_{VF} is the cost of expanding q^{\otimes} on a vertical architecture and C_R is the computational cost for searching the full index with the final query. The C_{VF} efficiency measure proposed in Aly et al. [1] accounts for two separate costs:

$$C_{VF}(q^{\otimes}, D_V) = C_{SEL}(q^{\otimes}, D_V) + C_{VR}(q^{\otimes}, D_V) \quad (7)$$

where $C_{SEL}(q^{\otimes}, D_{CSI})$ is the cost of the resource selection algorithm, and $C_{VR}(q^{\otimes}, D_V)$ (defined later) is the cost of retrieving documents in parallel from the selected verticals $sel(q^{\otimes})$. The cost of resource selection $C_{SEL}(q^{\otimes})$ depends on the type of the resource selection algorithm used:

$$C_{SEL}(q^{\otimes}) = \begin{cases} CSI(q^{\otimes}) & \text{if sample-document} \\ V & \text{if vocabulary-based} \end{cases} \quad (8)$$

where V is the total number of verticals and $CSI(q^{\otimes}) = C_R(q^{\otimes}, D_{CSI})$ the number of postings accessed in the CSI for all the query terms in q^{\otimes} considering a sample-document resource selection algorithm. In the vocabulary-based resource selection algorithms [1], typically since a single look-up operation is performed, it is set to the total number of verticals $C_{SEL}(q^{\otimes}) = |D_V|$.

Definition 3.2 (Parallel retrieval cost). In a vertical search scenario, C_R is calculated for a given query q^{\otimes} using the number of postings that the vertical search has to access as follows:

$$\begin{aligned} C_{VR}(q^{\otimes}, D_V) &= \sum_{i=1}^{|sel(q^{\otimes})|} C_R(q^{\otimes}, D_{V_{m(i)}}) \\ &= \sum_{i=1}^{|sel(q^{\otimes})|} \sum_{t \in q^{\otimes}} postings_{D_{V_{m(i)}}}(t) \end{aligned} \quad (9)$$

where $sel(q^{\otimes})$ are the verticals selected by a resource selection algorithm and $C_R(q^{\otimes}, D_{V_{m(i)}})$ is the number of accessed postings in vertical i for all the terms in the initial query q^{\otimes} .

3.3.2 Query response latency. A federated search architecture also affords faster response times via parallel work since multiple verticals can be searched in parallel. Considering the set of documents D , the latency metric C_{Lat} employed by Kulkarni and Callan [13], quantifies the longest execution path C_L for a given query q^{\otimes} , assuming a distributed query processing framework,

$$\begin{aligned} C_{Lat}(q^{\otimes}, D_V) &= C_{SEL}(q^{\otimes}, D_V) + C_L(q^{\otimes}, D_V) \\ &= C_{SEL}(q^{\otimes}, D_V) + \max_{i=1}^{|N|} \sum_{t \in q^{\otimes}} postings_{D_{V_{m(i)}}}(t) \end{aligned} \quad (10)$$

where $postings_{D_{V_{m(i)}}}(t)$ is the number of accessed postings in the inverted index for term t in vertical i , for a total of N verticals in a federated search environment.

3.4 Costs Comparison

The computational cost of the PRVF approach is strongly correlated to the cost of retrieving candidate documents in a federated search retrieval system. There might be non-negligible differences in the cost of the queries generated using different corpora for expansion. That said, if the number of terms in the expanded query is fixed for all methods, the cost of retrieving the final results $C_R(q)$ can be assumed to be of the same order of magnitude for all methods presented. Therefore, the first part of the cost equations Eq. (2) and Eq. (6), i.e., the cost of the query expansion process, C_{QE} , can be used alone to compare both approaches in terms of average computational costs:

$$C_{VF}(q^{\otimes}, D_{VF}) \ll C_R(q^{\otimes}, D) \quad (11)$$

$$\sum_{i=1}^{|sel(q^{\otimes})|} \sum_{t \in q^{\otimes}} postings_{D_{V_{m(i)}}}(t) \ll \sum_{t \in q} postings(t) \quad (12)$$

The cost of the initial retrieval when using the whole collection is much larger than the proposed alternatives. Using an index built with the posts of news sources provides a high-quality coverage of microblog user interests. Further computational gains are obtained by organizing news sources into topical verticals. The hypothesis is that a PRVF offers the lowest query expansion computational cost, and can provide comparable retrieval effectiveness to standard PRF techniques that use the whole corpus for feedback. Experiments will now examine this hypothesis.

4 EXPERIMENTAL METHODOLOGY

4.1 Datasets

4.1.1 Microblog datasets. We use the Tweets2013 corpus and the topics from the TREC 2013 and TREC 2014 Microblog track [17]. Tweets2013 is a microblog posts collection (approx. 240 million tweets) created by listening via Twitter's streaming API over the period: 1 February, 2013 – 31 March, 2013 (inclusive). NIST provided relevance judgments on a three-point scale of "informativeness": not relevant, relevant and highly relevant.

4.1.2 Vertical Expansion Corpus. Informed by previous work [22] and the categories presented in Twitter's sign up process, we created the following verticals: *general, politics, entertainment, technology, breaking, movies, science, sports, music* and assigned each source to a vertical as shown in Table 1. To build this corpus we selected 70 accounts from reliable news sources on Twitter. We selected accounts from publications that are reputable and that are also popular on Twitter (i.e. have a high number of followers).

We obtained the set of posts published² by crawling the timelines of these news sources on the period covered by the Tweets2013 corpus: 1 February, 2013 – 31 March, 2013 (inclusive), therefore we named it *NewsSources* (140,087 documents). A smaller collection (16,687 documents) is used as CSI for the evaluation of the resource selection algorithms and PRVF. The collection was crawled using Twitter's ~1% and ~10% samples simultaneously. Documents from both streams were added to the index, which resulted in a volume of documents that is about 12% of the size of *NewsSources*.

²<http://datasets.novasearch.org/tweets2013-newssources/>

Table 1: Verticals and sources.

Vertical (D_{V_i})	Twitter accounts
general	abc, ap, bbcnews, bbcworld, cbsnews, cnn, cnni, foxnews, huffingtonpost, latimes, nprnews, nytimes, reuters, reutersuk, usatoday, mashable
politics	huffpostpol, politico, theeconomist, washingtonpost, wsj
technology	arstechnica, cnet, gizmodo, techcrunch, wired, wireduk, thenextweb, techrepublic, cnet, gigaom, macworld
sports	bbcspport, sinow, eurosport, eu-rospportukt, sportscenter, espn
music	clash_music, rollingstone, nme, spin-magazine, stereogum, billboard, altpress, pitchfork
movies	americancine, thr, nytmovies, bbcfilms, totalfilm, guardianfilm, backstage, empiremagazine, filmcomment, timeout-film, sightssoundmag
entertainment	time, ew, variety, vanityfair, uncut-magazine
science	livescience, popsci, wiredscience, nasa, natgeo, newscientist
breaking	bbcbreaking, breakingnews, cnnbrk

PRVF (taily) selected 2.19 verticals on average for both TREC 2013 and TREC 2014 query sets. PRVF (ranks) selected 2.22 and 1.81 verticals on average for the TREC 2013 and TREC 2014 queries, respectively.

4.2 Retrieval Methods

To compare PRVF to previous research we considered 1 method without query expansion, 2 methods with expansion on the main corpus, and 5 methods with external expansion corpus. Performance was assessed in terms of MAP, NDCG@30, recall and computational cost (retrieval cost and latency).

No-PRF is a *query-likelihood* retrieval model with Dirichlet smoothing ($\mu = 2500$).

PRF is a standard pseudo-relevance feedback method that uses the whole search index for feedback. The RM3 pseudo-relevance feedback algorithm [16] is used for all the methods based on PRF because it was shown to be very effective in previous microblog retrieval research and it has similar information requirements and computational characteristics to other PRF algorithms. In all the PRF based methods, for feedback, we use the top 50 documents retrieved for each query q using the *query-likelihood* retrieval model, language modeling with Dirichlet smoothing ($\mu = 2500$). For PRF.wiki the number of documents used for feedback was reduced to 10 articles. The top-retrieved documents are then used to generate an expanded query q^e with a length of 20 terms. The expansion terms q^e are interpolated with the original query terms q^\emptyset with equal weight ($\lambda = 0.5$).

CLRM (Condensed List Relevance Models) is an approach, based on relevance models, recently proposed by Diaz [8] that essentially re-ranks the list of results retrieved by the initial query using the expanded query generated with the same list.

PRF.wiki is a pseudo-relevance feedback baseline that uses an external index of the English Wikipedia article pages. It was used for expansion in blog search by Arguello et al. [2] with significant improvements in effectiveness. We process a Wikipedia dump³, which dates from just before the microblog evaluation dataset, using *wikiextractor*⁴ to obtain clear indexable text.

SPRF is a pseudo-relevance feedback baseline that uses the whole *NewsSources* dataset as an external expansion corpus.

PRVF (taily) can adjust the number of selected verticals $|sel(q^\emptyset)|$ dynamically for each query. It was parameterized with the values ($n = 400$ and $v = 50$) recommended by Aly et al. [1].

PRVF (ranks) also adjusts the number of selected verticals $|sel(q^\emptyset)|$ dynamically for each query. To limit the number of verticals similarly for Rank-S we set the threshold $minRanks = 1e^{-6}$.

PRVF (cracs) inspects a fixed number of verticals: $|sel(q^\emptyset)| = \{1, 2, 3\}$. For instance, PRVF (cracs1) corresponds to expanding the queries using only the top vertical selected by CRCS.

5 RESULTS AND DISCUSSION

The evaluation is organized as follows: we start by analyzing the efficiency (Section 5.1), biases (Section 5.2) and effectiveness (Section 5.3) of PRVF and SPRF methods. Lastly, we compare the standard implementation of PRF based on re-retrieval with the recently proposed implementation Condensed List Relevance Models (CLRM) [8] based on re-ranking (Section 5.4).

5.1 Cost Analysis of PRF Methods

In this paper we focus on computational cost reduction for the initial retrieval, necessary in PRF-based query expansion. In this section we measure the computational cost, C_{QE} , as the number of accessed posting lists for query expansion. See Section 3.4 for the query expansion costs of each method. Figure 1 shows the trade-offs between the cost, C_{QE} , and the corresponding results on retrieval metrics on the TREC Microblog datasets. C_{QE} is represented in the y-axis in log-scale to get an overview of how PRVF compares to the No-PRF and standard PRF baselines. The x-axis is represents either the MAP or the NDCG@30 retrieval metric. Since the objective is to lower C_{QE} and get better retrieval precision, the desired method would fall below the dashed line that goes from No-PRF to PRF, towards the bottom right corner.

The PRVF methods have always a lower computational cost C_{QE} than SPRF, clustering just below it in the graphs. Even though the cost of the PRVF methods is considerably lower, the MAP results obtained are near those of SPRF. Therefore, SPRF can be a good predictor for the expected retrieval effectiveness for the PRVF architecture (see Figure 1b).

While the initial expectation is that more computational cost should translate into a better ranking, some PRVF methods provide a better top rank with a lower computational cost. The NDCG@30 results obtained with PRVF methods are similar or better than PRF (see Figure 1d). PRVF (taily) outperformed the other methods in NDCG@30 in the TREC 2014 queries. However, on the TREC 2013 queries PRVF (cracs3) outperformed the PRVF (taily) method slightly as it can be seen in Figure 1c.

³enwiki-20130102-pages-articles.xml

⁴<https://github.com/attardi/wikiextractor>

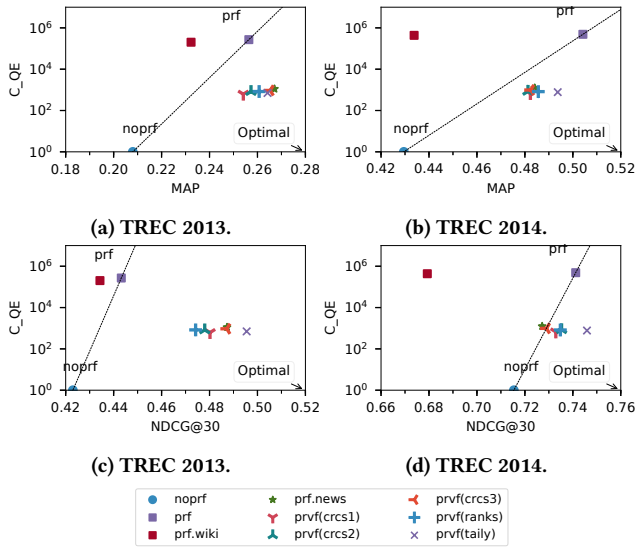


Figure 1: Comparison of retrieval cost (C_{QE}) versus retrieval effectiveness (using precision metrics MAP and NDCG@30).

All the proposed approaches were three orders of magnitude less computationally expensive than the PRF baseline. The PRVF-based methods are the most efficient since for each query they search only the verticals that are most promising. We found that query expansion response times can be halved on both datasets compared to the SPRF baseline as measured by C_{Lat} , which takes into account the parallelism afforded by the distributed architecture.

5.2 Quality of Expansion Corpus

In this section we analyze potential biases in the vertical expansion corpus and the importance of the expansion corpus age and time span. Since PRVF uses documents from news sources for query expansion we might improve the chances of retrieving tweets from these sources. To make sure that bias is not improving the results unfairly we counted the number of documents marked as relevant in the main index (TREC 2013 and 2014) that are in the expansion corpus (NewsSources). The overlap was only 9 relevant documents in TREC 2013 and 13 relevant documents for TREC 2014. Thus, we did not find any evidence that the choice of news sources affords any kind of unfair advantage.

A key aspect of the PRVF architecture is its ability to cope with multiple information streams that are constantly feeding the query expansion corpus. In Figure 2a and Figure 2b we observe how the expansion corpus age, i.e. the difference between queries timestamp and the most recent document timestamp, is clearly linked to the decay in retrieval precision. The time span of the expansion corpus is also examined in Figure 2c and 2d – here we can observe that it might be sufficient to keep only the last 15 days for query expansion. This confirms the initial assumption that in microblog search it is critical to use an up-to-date expansion corpus. In addition, we found that it does not seem to be necessary to keep an expansion corpus with a long time span to answer most queries effectively.

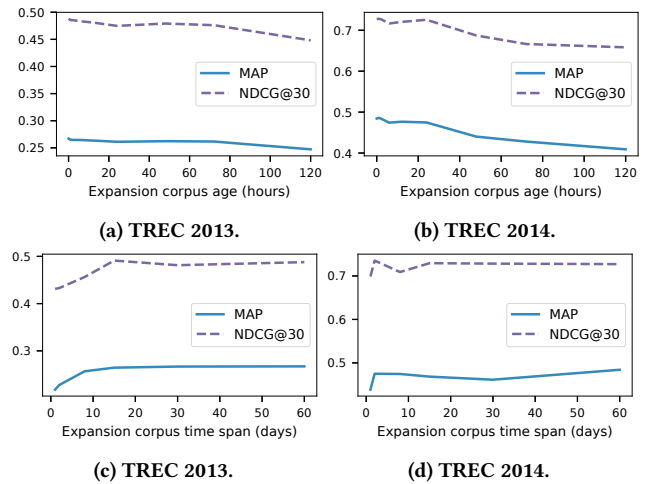


Figure 2: Analysis of expansion corpus age and time span.

5.3 Retrieval Effectiveness of PRF Methods

We show the detailed results of our evaluation on Table 2 and Table 3. We present the average over all queries for the ranking metrics MAP and NDCG@30. The average computational cost of each approach is presented in the C_{QE} column. After the C_{QE} , in parentheses, we show cost reduction in relation to SPRF. The language model baseline, No-PRF, does not use pseudo-relevance feedback therefore C_{QE} does not apply.

A federated query expansion architecture also affords faster response times via parallel work. When multiple verticals are searched in parallel, the query expansion process waits for all the response of all verticals to proceed with the term selection phrase. Therefore we measure response time using C_{Lat} which can give us the maximum amount of work done by any vertical searched in parallel. We find that query expansion times are halved in both datasets compared to the SPRF baseline.

The PRF baseline improved the retrieval effectiveness metrics considerably over No-PRF and PRF.wiki. In the TREC 2013 queries the PRF baseline improved on MAP over No-PRF by 23.3% and NDCG@30 by 4.8%, and the MAP improvement was statistically significant. For TREC 2014 queries PRF improved on MAP by 17.4% and NDCG@30 by 3.6% over No-PRF, and the MAP improvement was statistically significant. However, the average cost of the query expansion process using standard PRF for TREC 2013 and TREC 2014 was very high at 269k and 488k postings, respectively.

We followed onto expansion methods with external corpus. Using a recent Wikipedia corpus for feedback, PRF.wiki, only provided an improvement of MAP on the TREC 2013 queries. However, for the TREC 2014 queries, NDCG@30 was 5% lower than the No-PRF baseline. In addition, even though the Wikipedia corpus is smaller than the search corpus the average computational costs were still very high at 202k and 434k accessed postings for TREC 2013 and TREC 2014, respectively. We conclude then that using Wikipedia for query expansion in microblogs can harm retrieval effectiveness when the expansion collection is not up-to-date.

Table 2: TREC 2013 dataset results.

	C_{Lat}	C_{QE}	MAP	NDCG
w/o External Expansion Corpus				
No-PRF	0	0	0.2080	0.4230
PRF	269175	269175	0.2564	0.4432
w/ External Expansion Corpus				
PRF.wiki	201892	201892	0.2323	0.4343
SPRF	1110	1110	0.2671 [‡]	0.4873 [‡]
w/ External Vertical Expansion Corpus				
PRVF (crs1)	631 -43.2%	631 -43.2%	0.2541 [‡]	0.4802 [‡]
PRVF (crs2)	653 -41.2%	845 -23.9%	0.2573 [‡]	0.4780 [‡]
PRVF (crs3)	655 -41.0%	956 -13.9%	0.2653 [‡]	0.4872 [‡]
PRVF (ranks)	673 -39.4%	903 -18.6%	0.2607 [‡]	0.4742 [‡]
PRVF (taily)	509 -54.1%	703 -36.7%	0.2642 [‡]	0.4955 [‡]

Symbols [†] and [‡] stand for a statistically non-inferior result to PRF with $p < 0.05$ and $p < 0.01$ respectively, according to a non-inferiority test [28].

Using the *NewsSources* corpus for feedback, PRF.news, provided a significant improvement in terms of retrieval precision and computational cost. However, to fully verify the aforementioned hypothesis, we examined the impact of creating expansion verticals.

In the group of methods that use an external vertical expansion corpus, the PRVF (taily) method was the most balanced in the TREC 2013 queries with a C_{QE} of only 703, which corresponds to a cost reduction of 36.7% over SPRF, around 2.2× faster. It had one of the highest MAP results (3.0% higher than PRF) and improved 27.0% over No-PRF (statistically significant). It also had the second best NDCG@30 result, improving 1.7% over SPRF and 11.8% over the standard PRF approach.

PRVF (taily) provided the best balance for the TREC 2014 queries as well. It obtained a 14.7% improvement in MAP over No-PRF (statistically significant) and a 4.4% improvement in NDCG@30 with a computational cost of only $C_{QE} = 773$. PRVF (taily) was around 2.2× faster than searching the whole news index SPRF a cost reduction of 37.6%. The highest MAP results for TREC 2013 were obtained with PRVF (crs3). Because a fixed number of verticals are searched for each query (3), which is higher than PRVF (taily)'s average, the cost reduction is smaller (13.9% over SPRF). Last, but not least, PRVF (taily) was the fastest method, delivering the lowest latency (C_{Lat} column), halving the latency of PRF.news.

5.4 Re-ranking PRF and Short Text Documents

In Table 4 we present retrieval effectiveness metrics for CLRM and other PRF implementations based on re-retrieval. We also present the *recall* metric on the top 1000 results (number of relevant @ 1000/total of relevant). We found that CLRM always outperformed No-PRF in MAP and NDCG@30. However, since CLRM does not perform a re-retrieval it re-ranks the documents retrieved by the initial query, therefore its recall is equal to the query likelihood baseline No-PRF. Due to this, CLRM is not able to achieve the same

Table 3: TREC 2014 dataset results.

	C_{Lat}	C_{QE}	MAP	NDCG
w/o External Expansion Corpus				
No-PRF	0	0	0.4295	0.7154
PRF	487547	487547	0.5042	0.7412
w/ External Expansion Corpus				
PRF.wiki	434233	434233	0.4338	0.6793
SPRF	1239	1239	0.4841	0.7272 [†]
w/ External Vertical Expansion Corpus				
PRVF (crs1)	661 -46.7%	661 -46.7%	0.4823	0.7329 [‡]
PRVF (crs2)	734 -40.8%	848 -31.6%	0.4813	0.7353 [‡]
PRVF (crs3)	734 -40.8%	982 -20.7%	0.4824	0.7290 [†]
PRVF (ranks)	734 -40.8%	821 -33.7%	0.4856	0.7348 [‡]
PRVF (taily)	575 -53.6%	773 -37.6%	0.4927 [‡]	0.7470 [‡]

Symbols [†] and [‡] stand for a statistically non-inferior result to PRF with $p < 0.05$ and $p < 0.01$ respectively, according to a non-inferiority test [28].

Table 4: Retrieval results using CLRM on microblog datasets.

	TREC 2013			TREC 2014		
	MAP	NDCG	Recall	MAP	NDCG	Recall
No-PRF	0.2080	0.4230	0.5188	0.4295	0.7154	0.6994
PRF	0.2564	0.4432	0.5764	0.5042	0.7412	0.7860
CLRM	0.2276	0.4423	0.5188	0.4718	0.7416	0.6994
PRF.wiki	0.2323	0.4343	0.5689	0.4338	0.6793	0.7443
PRVF (taily)	0.2642	0.4955	0.5921	0.4927	0.7470	0.7818

level of retrieval effectiveness of the implementations based on re-retrieval. Even though the generated expanded query is the same for CLRM and PRF, PRF was more effective.

In short-text document indexes some relevant documents that are ranked at the top by a re-retrieval implementation might be missing from the initial retrieval using the original query terms only. In addition, some relevant documents might contain only a few of the original query terms, a problem that is exacerbated by the short size of the documents in a microblog corpus. Therefore, in short text datasets an implementation of pseudo-relevance feedback based on re-retrieval might be preferred to achieve similar retrieval effectiveness to standard PRF.

The PRF.wiki method, based on re-retrieval, was able to retrieve more relevant documents than CLRM with a higher recall in both datasets. However, this higher recall did not translate into better results since the query expansions generated from Wikipedia were less effective for ranking.

PRVF (taily) approach generates query expansions using a more efficient federated query expansion architecture over an external news corpus. The quality of the expansions generated by this method can be attested from its high recall and better precision than the standard PRF approach.

6 CONCLUSION

In this paper, we studied an efficient method for pseudo-relevance feedback that organizes large collections of documents, published by a set of news sources into news verticals. The evaluation of the proposed architecture led us to the following concluding points.

Federated QE. PRVF architecture is an efficient federated QE architecture for microblog search, where the expansion corpus is live and has new documents arriving from news sources in a streaming fashion. This approach outperformed the retrieval effectiveness of using the non-partitioned news index (SPRF) and standard PRF.

Low-cost and effective PRF. The best balance between efficiency and effectiveness was obtained using PRVF (taily), which was relatively more robust than other approaches while using on average fewer verticals. PRVF (taily) achieved the highest results in effectiveness metrics for both the TREC 2013 and TREC 2014 query sets. PRVF (csrcs3) was the best in terms of MAP on TREC 2013, but at a slightly higher computational cost. This indicates that resource selection algorithms that can dynamically limit the number of verticals searched are more suitable for this task.

Quality of the query expansion corpus. Understanding the properties of the search domain is key to ensure the quality of the expansion corpus. In microblog search, news sources as the ones in table 1 guarantee a good and up-to-date coverage of user interests. This is a critical aspect that is addressed by domain-specific knowledge.

ACKNOWLEDGMENTS

This work has been partially funded by the CMU Portugal research project GoLocal Ref. CMUP-ERI/TIC/0033/2014, by the H2020 ICT project COGNITUS with the grant agreement n° 687605, and by the FCT project NOVA LINC3 Ref. UID/CEC/04516/2013.

REFERENCES

- [1] Robin Aly, Djoerd Hiemstra, and Thomas Demeester. 2013. Taily: Shard Selection Using the Tail of Score Distributions. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '13)*. ACM, New York, NY, USA, 673–682.
- [2] J. Arguello, J.L. Elsas, J. Callan, and J.G. Carbonell. 2008. Document Representation and Query Expansion Models for Blog Recommendation. In *Proceedings of the 2nd International Conference on Weblogs and Social Media (ICWSM '08)*. 10–18.
- [3] Ricardo Baeza-Yates, Vanessa Murdock, and Claudia Hauff. 2009. Efficiency Trade-Offs in Two-Tier Web Search Systems. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. ACM, New York, NY, USA, 163–170.
- [4] Michael Bendersky, Donald Metzler, and W. Bruce Croft. 2012. Effective Query Formulation with Multiple Information Sources. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM '12)*. ACM, New York, NY, USA, 443–452.
- [5] Marc-Allen Cartright, James Allan, Victor Lavrenko, and Andrew McGregor. 2010. Fast Query Expansion Using Approximations of Relevance Models. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM '10)*. ACM, New York, NY, USA, 1573–1576.
- [6] Jaeho Choi and W. Bruce Croft. 2012. Temporal Models for Microblogs. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, USA, 2491–2494.
- [7] Thomas Demeester, Dolf Trieschnigg, Dong Nguyen, and Djoerd Hiemstra. 2013. Overview of the TREC 2013 Federated Web Search Track. In *Proceedings of The Twenty-Second Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*, Ellen M. Voorhees (Ed.), Vol. Special Publication 500-302. National Institute of Standards and Technology (NIST).
- [8] Fernando Diaz. 2015. Condensed List Relevance Models. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval (ICTIR '15)*. ACM, New York, NY, USA, 313–316.
- [9] Fernando Diaz and Donald Metzler. 2006. Improving the Estimation of Relevance Models Using Large External Corpora. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. ACM, New York, NY, USA, 154–161.
- [10] Miles Efron. 2010. Hashtag Retrieval in a Microblogging Environment. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '10)*. ACM, New York, NY, USA, 787–788.
- [11] Feifan Fan, Runwei Qiang, Chao Lv, and Jianwu Yang. 2015. Improving Microblog Retrieval with Feedback Entity Model. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM '15)*. ACM, New York, NY, USA, 573–582.
- [12] Fatih Hafizoglu, Emre Can Kucukoglu, and Ismail Sengor Altinogvde. 2017. On the Efficiency of Selective Search. In *Advances in Information Retrieval*. Springer, Cham, 705–712.
- [13] Anagha Kulkarni and Jamie Callan. 2015. Selective Search: Efficient and Effective Search of Large Textual Collections. *ACM Trans. Inf. Syst.* 33, 4 (April 2015), 17:1–17:33.
- [14] Anagha Kulkarni, Almer S. Tigelaar, Djoerd Hiemstra, and Jamie Callan. 2012. Shard Ranking and Cutoff Estimation for Topically Partitioned Collections. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, USA, 555–564.
- [15] Victor Lavrenko and James Allan. 2006. *Real-Time Query Expansion in Relevance Models*. IR 473. University of Massachusetts Amherst.
- [16] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. ACM, New York, NY, USA, 120–127.
- [17] Jimmy Lin and Miles Efron. 2013. Overview of the TREC-2013 Microblog Track. In *Proceedings of The Twenty-Second Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*, Ellen M. Voorhees (Ed.), Vol. Special Publication 500-302. National Institute of Standards and Technology (NIST).
- [18] Craig Macdonald, Nicola Tonello, and Iadh Ounis. 2012. Learning to Predict Response Times for Online Query Scheduling. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. ACM, New York, NY, USA, 621–630.
- [19] Kamran Massoudi, Manos Tsagkias, Maarten de Rijke, and Wouter Weerkamp. 2011. Incorporating Query Expansion and Quality Indicators in Searching Microblog Posts. In *Advances in Information Retrieval (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 362–367.
- [20] Alistair Moffat, William Webber, Justin Zobel, and Ricardo Baeza-Yates. 2007. A Pipelined Architecture for Distributed Text Query Evaluation. *Information Retrieval* 10, 3 (June 2007), 205–231.
- [21] Paul Ogilvie and Jamie Callan. 2001. The Effectiveness of Query Expansion for Distributed Information Retrieval. In *Proceedings of the Tenth International Conference on Information and Knowledge Management (CIKM '01)*. ACM, New York, NY, USA, 183–190.
- [22] Kevin Dela Rosa, Rushin Shah, Bo Lin, Anatole Gershman, and Robert Frederking. 2011. Topical Clustering of Tweets. In *Proceedings of the ACM SIGIR: SWSM*.
- [23] Milad Shokouhi. 2007. Central-Rank-Based Collection Selection in Uncooperative Distributed Information Retrieval. In *Advances in Information Retrieval (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 160–172.
- [24] Milad Shokouhi, Leif Azzopardi, and Paul Thomas. 2009. Effective Query Expansion for Federated Search. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. ACM, New York, NY, USA, 427–434.
- [25] Milad Shokouhi and Luo Si. 2011. Federated Search. *Found. Trends Inf. Retr.* 5, 1 (Jan. 2011), 1–102.
- [26] Luo Si and Jamie Callan. 2003. Relevant Document Distribution Estimation Method for Resource Selection. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '03)*. ACM, New York, NY, USA, 298–305.
- [27] Jaime Teevan, Daniel Ramage, and Merredith Ringel Morris. 2011. #TwitterSearch: A Comparison of Microblog Search and Web Search. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM '11)*. ACM, New York, NY, USA, 35–44.
- [28] Esteban Walker and Amy S. Nowacki. 2011. Understanding Equivalence and Noninferiority Testing. *Journal of General Internal Medicine* 26, 2 (Feb. 2011), 192–196.
- [29] Wouter Weerkamp, Krisztian Balog, and Maarten de Rijke. 2012. Exploiting External Collections for Query Expansion. *ACM Trans. Web* 6, 4 (Nov. 2012), 18:1–18:29.
- [30] Ryan W. White and Gary Marchionini. 2007. Examining the Effectiveness of Real-Time Query Expansion. *Inf. Process. Manage.* 43, 3 (May 2007), 685–704.
- [31] Stewart Whiting, Iraklis A. Klampanos, and Joemon M. Jose. 2012. Temporal Pseudo-Relevance Feedback in Microblog Retrieval. In *Advances in Information Retrieval (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 522–526.
- [32] Yang Xu, Gareth J.F. Jones, and Bin Wang. 2009. Query Dependent Pseudo-Relevance Feedback Based on Wikipedia. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. ACM, New York, NY, USA, 59–66.