Rank-Energy Selective Query Forwarding for Distributed Search Systems

Amin Teymorian
Georgetown University
Washington, DC, USA
amin@cs.georgetown.edu

Ophir Frieder
Georgetown University
Washington, DC, USA
ophir@cs.georgetown.edu

Marcus A. Maloof
Georgetown University
Washington, DC, USA
maloof@cs.georgetown.edu

ABSTRACT

Scaling high-quality, cost-efficient query evaluation is critical to search system performance. Although partial indexes reduce query processing times, result quality may be jeopardized due to exclusion of relevant non-local documents. Selectively forwarding queries between geographically distributed search sites may help. The basic idea of query forwarding is that after a local site receives a query, it determines non-local sites to forward the query to and returns an aggregation of the local and non-local results. Nevertheless, electricity costs remain substantial sources of operating expenses. We present a hybrid rank-energy query forwarding model termed "RESQ." The novel contribution is to simultaneously consider both ranking quality and spatially-temporally varying energy prices when making forwarding decisions. Experiments with a large-scale query log, publicly-available electricity price data, and real search site locations demonstrate that query forwarding under RESQ achieves the result scalability of partial indexes with the cost savings of energy-aware approaches (e.g., an 87% ranking guarantee with a 46% savings in energy costs).

Categories and Subject Descriptors

H.3.3 [**Information Storage Systems**]: Information Retrieval System

Keywords

Distributed IR, rank, energy, query forwarding, linear programming

1. INTRODUCTION

Modern search systems operate under an unprecedented amount of content, traffic volume, and user distribution. For example, Google indexes $\approx$ 20 billion pages per day [17] and processes over 4 billion queries per day [12], while YouTube has $\approx$ 72 hours of video uploaded each minute, and 70% of its traffic originating from outside the USA [38]. Therefore,

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2505710.

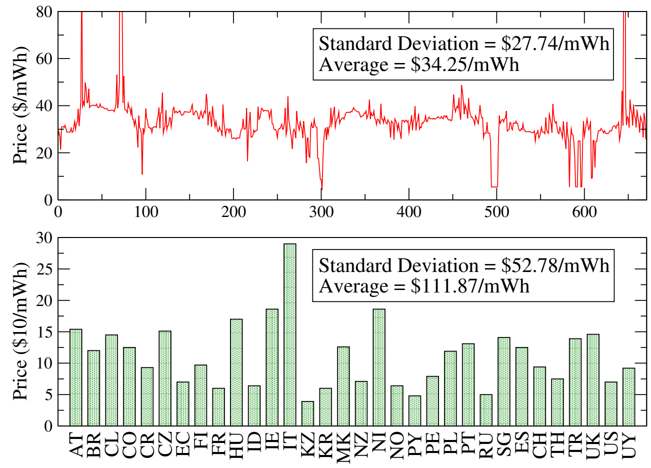


Figure 1: Energy price temporal and spatial variation: (top) NYISO energy prices [29], 15 minute intervals; (bottom) EIA energy prices by country [37].

scaling high-quality, cost-effective query evaluation is a critical challenge to search system performance. Our solution is RESQ (pronounced "rescue"). Introduced in our poster at ACM CIKM '12 [36], we generalize RESQ to query volumes across realistic search site locations under an explicit latency model. The novel idea of RESQ is to selectively forward queries between sites in a distributed search system such that result quality and spatial- and temporal-variations in energy costs (e.g., the prices illustrated in Fig. 1 and elaborated on below) are optimized jointly.

In a distributed search system [2], selectively forwarding queries between the geographically-distant search sites has been shown to improve scalability [8]. For example, query evaluation can be faster over smaller subsets or "partitions" of full indexes. The underlying intuition is that well-localized distributed smaller indexes can yield faster query evaluation times and overall cost savings in spite of additional network latency incurred from forwarding queries that target non-local documents [9]. The idea of "different results from different sites" is essential to query forwarding (e.g., [2], [9], and [23]). However, in practice, there is no restriction on the "local" and "non-local" partitioning strategy employed (e.g., full index replication [24]). As discussed in Sec. 2 and Sec. 3, RESQ forwarding supports such partitionings while returning high-quality results and reducing operating costs.

Scalability may also be considered in contexts other than query evaluation times (e.g., energy costs [33] and [24]). For example, the monetary cost of evaluating a volume of queries may be considered. While such costs can be attributed to a number of sources (e.g., labor), an important consideration for search system cost-effectiveness is data center electricity costs (e.g., [30]). Interestingly, although well-known search systems spend an estimated $10s of millions each year on electricity, the costs may be substantially decreased by exploiting spatial and temporal variations in energy prices (e.g., variations due to daytime demand) [33]. In the context of distributed search systems, the basic idea is that among multiple data center locations, the query workload is shifted to the sites with cheaper electricity prices [24]. As elaborated on below, the RESQ forwarding model not only explicitly considers energy prices, but also simultaneously optimizes for result quality independent of the underlying partitioning strategy (cf. assuming full index replication [24]).

Our contribution, RESQ, is a hybrid rank-energy selective query forwarding model that obtains the scalability of index-partitioned systems and the cost savings of energy-aware approaches. Our idea is to selectively forward queries such that the result ranking guarantee (see Sec. 3) in a non-replicated, geographically-distributed search system is maximized given a budget for spatially- and temporally-varying energy prices, such as those illustrated in Fig. 1. Besides a novel simultaneous consideration of result quality and energy costs, RESQ's design complements existing work well (e.g., we support dynamic energy prices, cf. free of explicit energy consideration; and we also support partitioned or "non-replicated" indexes, cf. fully-replicated indexes). Experiments with query and electricity data demonstrate the merit of RESQ's balanced approach (e.g., an 87% ranking guarantee with 46% energy savings). The unique main contribution of our study is two-fold:

- We formalize the first selective query forwarding model that directly considers both result quality and energy prices in a distributed search system with partitioned indexes.

- We design, implement, and empirically evaluate a distributed algorithm that instantiates the model, and demonstrate its ability to obtain favorable tradeoffs between possible result quality and energy expenses.

In the sections that follow, we describe related research efforts (Sec. 2), introduce RESQ forwarding (Sec. 3), evaluate its performance (Sec. 4), and summarize the study and discuss future research questions (Sec. 5).

2. RELATED EFFORTS

Distributed information retrieval (IR) is of substantial interest to the IR research community (e.g., [2–4, 6], [8], [9], [23], [24], and [26–28]). Although traditionally focused on efficiency within a search site, such as by partitioning a document collection into smaller collections or "shards" (e.g., [26–28]), recent research has demonstrated the feasibility of distributed IR at the system level (e.g., [2], [4], [8], [9], [23], and [24]).

Critical to achieving distributed search system performance is selective query forwarding as not all documents will be locally indexed. Selecting the sites that a query should be forwarded to in a distributed fashion is a non-trivial problem. Indeed, there has been a number of recent research efforts along the query forwarding front (e.g., [2], [8], [9], [23], and [24]).

Table 1: A summary of recent approaches to selective query forwarding, including indicators for design properties of the underlying search system (i.e., index partitioning and energy cost awareness).

Approach	Property	
	Index Partition	Energy-Aware
Baeza-Yates *et al.* [2]	✓	
Cambazoglu *et al.* [9]	✓	
Kayaaslan *et al.* [24]		✓
Junqueira *et al.* [23]	✓	

Cambazoglu *et al.* [9] increase forwarding efficiency when partial (or "non-replicated") indexes are employed by locally computing an upper bound on the maximum possible result ranking of a non-local result for a locally-received query. Based on offline results from non-local indexes, a local site only forwards a query to sites that might have a result that would rank within its top-k results. The thresholding strategy generalizes an earlier thresholding approach by Baeza-Yates *et al.* [2].

Besides contributing a sophisticated reactive index replication algorithm "RIP", Junqueira *et al.* [23] recently introduce a threshold-based query forwarding heuristic. The basic idea is to upper bound non-local documents with the average of exact partial scores or the lowest available score in the posting lists corresponding to the query terms. The scores are compared to those of documents that are indexed locally after initial assignment or replicated under a specified replication budget.

On the other hand, Kayaaslan *et al.* [24] selectively forward queries in a replicated search system (i.e., evaluations over full indexes) to reduce operating costs based on differences in site energy prices and available query processing capacity. Local sites forward queries to non-local sites with cheaper energy prices with probabilities proportional to the fraction of the local workload that the non-local site could process after evenly distributing its available capacity to sites with higher energy prices.

For our study, the properties of the aforementioned query forwarding algorithms that most motivate RESQ are summarized in the taxonomy illustrated in Table 1. Our goal is a selective query forwarding algorithm that avoids unnecessary forwards (e.g., [9]) and simultaneously saves energy costs (e.g., [24]). Specifically, we ask the following two questions:

- What does query forwarding designed to allow both the scalability of partial (or non-replicated) indexes (e.g., [9]) and consideration of spatial and temporal variations in energy prices look like (e.g., [24])?

- Given a feasible approach, what is the tradeoff between the possible ranking quality and energy savings? That is, given an approach, under what conditions is such an algorithm useful for distributed search systems?

We present our answer to the first question, called RESQ forwarding, next in Secion 3. To answer the second question, we conduct the empirical evaluation described in Sec. 4.

3. RANK-ENERGY SELECTIVE QUERY FORWARDING

We consider a distributed search system consisting of a set $S = \{S_1, \ldots, S_m\}$ of m sites distant geographically. An index I is partitioned disjointly among the sites; partial index I_j is assigned to site S_j, $I_j \cap I_k = \emptyset$ for $j \neq k$, and $\cup_{j=1}^{m} I_j = I$. Recall that other partitioning strategies may be employed as well. At a time t on continuous timeline T, a "local" site $\hat{S} \in S$ must evaluate a volume of n queries $Q = \{q_1, \ldots, q_n\}$. Note that a site is local (or "non-local") with respect to the site a query q_i is issued to originally. The objective for a local site $\hat{S}$ implementing RESQ forwarding is to select (not necessarily identical) subsets of non-local sites from $S \setminus \{\hat{S}\}$ to forward the queries $q_i \in Q$ to such that the sum of the non-local result ranking upperbounds r_j for evaluating q_i over I_j (or the "ranking guarantee") is as large as possible within a per-query energy price budget at the local site. For reference and readability the notation used is summarized in Table 2.

3.1 Linear Programming (LP) Formulation

Formulated as a linear programming (LP) problem, we maximize the sum of the non-local result-ranking upper-bounds (denoted "z"):

$$z = \sum_{i=1}^{n} \sum_{j=1}^{m} r_j(q_i) \times x_{ij}, \tag{1}$$

subject to a per-query, site-specific energy cost budget C_i:

$$\sum_{j=1}^{m} c_j(q_i) \times x_{ij} \leq C_i, \tag{2}$$

where $c_j(q_i) = \alpha_j \times |q_i| \times p_j(t)$ (and is explained below), and the feasibility constraints:

$$r_j(q_i) \geq 0, \; c_j(q_i) \geq 0, \text{ and } x_{ij} \in \{0, 1\}. \tag{3}$$

The function $c_j(\cdot)$ is the cost to evaluate a query q_i with $|q_i|$ terms over index I_j at site S_j where the spatially- and temporally-varying energy price at time instance t energy is $p_j(t)$. A site-specific constant α_j relates units of energy to query complexity. The indicator variable

$$x_{ij} = \begin{cases} 1 & \text{if query } q_i \text{ is evaluated at site } S_j; \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

A vector of x_{ij} values over all search sites S_j, $1 \leq j \leq m$, represents a solution for in RESQ forwarding. If x_{ij} is 1 (0) the results from S_j will (not) be included in the final result aggregation for query q_i originating at the local site $\hat{S}$.

In general, for $\hat{S}$ to obtain actual r_j values from non-local site S_j, the query is required to be forwarded and evaluated (i.e., all sites must evaluate the query, and the savings from selective query forwarding are obviated). Therefore, selective query forwarding models (such as [9]) employ a locally-computed upper bound on the non-local result rankings, denoted $\tilde{r}_j$. [1] We use r_j to refer to $\tilde{r}_j$ for simplified exposition. Also, local site $\hat{S}$ necessarily has $x_{ij} = 0$ because all non-local ranking estimates are relative to the results from evaluating q_i at $\hat{S}$.

[1]The intuition for $\tilde{r}_j$ is to avoid forwarding queries to sites with results guaranteed to rank outside the to-be-returned top-k set of documents (after aggregation at local site S_L).

Note that the price function $p_j(t)$ may consider non monetary costs as well (e.g., latency or load capacity). Indeed, the LP formulation is general enough to support additional constraints unrelated to ranking and energy. However, as defaulting to local evaluation is possible (e.g., [24], due to overflow), we limit c_j to electricity-related costs, and defer latency analysis to Sec. 4. The bound C_i is an independent per-query site-tunable parameter that RESQ only assumes to be in $\mathbb{R}^+$. Consistent with [24], we assume a stable energy price during forwarding and evaluation of an individual query. Lastly, the LP problem is amenable to existing techniques for combinatorial optimization. The total number of search sites also makes each problem instance quick to solve, and LP-based formalizations have appeared in other large scale systems research (e.g., [13], [39], and [9]).

3.2 Distributed Algorithm

Upon receiving a query volume Q, the (local) search site implements RESQ forwarding in an online distributed manner. Intuitively, selecting the (non-local) search sites to forward the queries to can be viewed as an iterative three-step process: for each query, compute the non-local query processing energy costs, locally bound the non-local result rankings, then optimally select the to-be-forwarded-to (non-local) sites based on a tradeoff between possible ranking quality and spatial-temporal energy costs.

Following our discussion in [36], the RESQ forwarding process is detailed in Algorithm 1, and an example is illustrated in Fig. 2. The algorithm begins when a local site (e.g., S_L in Fig. 2, left side) receives a query q_i (or set of queries Q) at time t. First, the local site receives the cost parameters p_j and α_j from all of the non-local sites (e.g., Line 2 in Algorithm 1); in Fig. 2, the sites are S_1 to S_8. The underlying assumption is that each site receives up-to-date electricity cost information (e.g., site-to-site broadcast or approximation from historical data [24]).

Algorithm 1 RESQ Forwarding

Require: (Q, t), the query set Q at time instance t.
1: **for** $j \in \{1, \ldots, |S|\}$ **do**
2: $\quad \{\alpha_j, p_j\} \leftarrow$ GETNONLOCALPARAMS(t)
3: **end for**
4: **for** $q_i \in Q$ **do**
5: $\quad \{c_j(q_i)\}_{j=1}^{|S|} \leftarrow$ UPDATECOSTS(q_i)
6: $\quad \{r_j(q_i)\}_{j=1}^{|S|} \leftarrow$ BOUNDRANKINGS(q_i)
7: $\quad \{x_{ij}\}_{j=1}^{|S|} \leftarrow$ LPSOLVE$(\{r_j, c_j\}_{j=1}^{|S|}, C_i)$
8: $\quad R = \emptyset$
9: $\quad$ **for** $S_j \in S$ **do**
10: $\quad\quad$ **if** $x_{ij} = 1$ **then**
11: $\quad\quad\quad \hat{R} \leftarrow$ FORWARD(q_i, S_j)
12: $\quad\quad\quad R \leftarrow$ MERGE$(\hat{R}, R)$
13: $\quad\quad$ **end if**
14: $\quad$ **end for**
15: $\quad$ RETURNTOPKDOCS(R)
16: **end for**

Next, the per-query site costs (i.e., the c_js), are updated using the costs parameters for time t (Line 5). After obtaining the costs, upper bounds on the quality of results that could be returned by non-local sites are locally computed (Line 6). Bounding helps eliminate "false-positive sites," i.e., non-local sites that cannot contribute results that rank

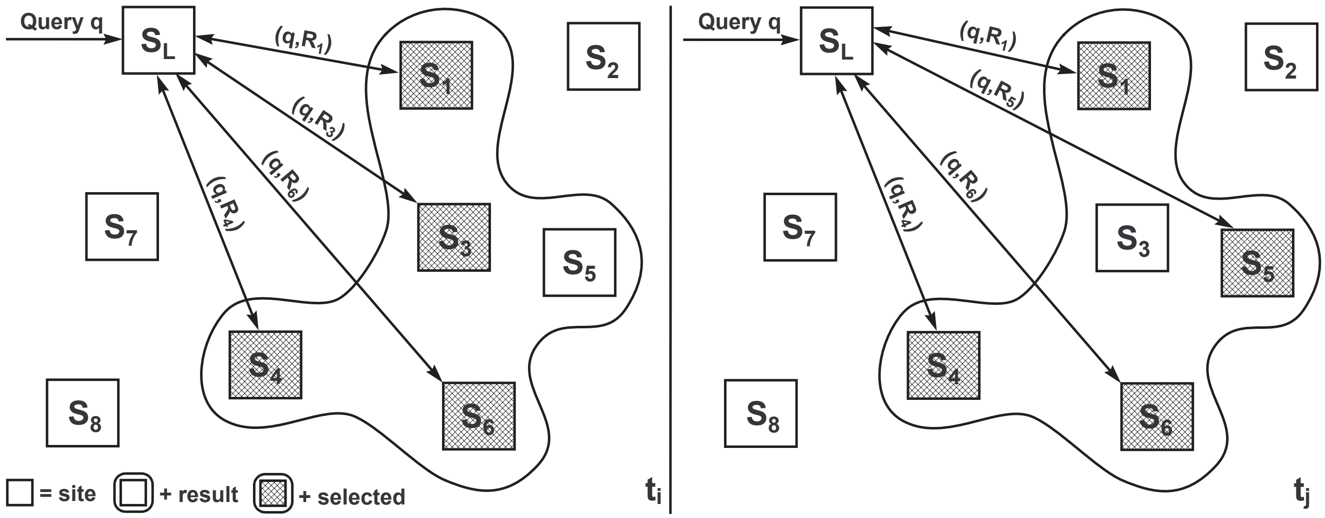


Figure 2: An illustration of RESQ forwarding. The basic idea is for the local S_L to forward the query to the subset of non-local sites (e.g., S_1, S_3, S_4, and S_5 at time t_i) that maximize the upperbounds on the non-local result quality given an energy budget. Note that at both time t_i and t_j the same query q is received at local site S_L. Although the collections of documents indexed at $S_1, \ldots, S_8$ also remains the same, a different subset of nodes are forwarded the query (e.g., S_3 at t_i is replaced with S_5 at t_j) due to the possible result quality given the combination of current energy price and budget C for the query (i.e., a new optimal solution is computed).

higher than those from the local index. For example, in Fig. 2, S_L computes upper bounds (via, for example, [9]) that imply that only S_1, S_3, S_4, S_5, and S_6 (i.e., the sites enclosed in the blob-like boundary) could possibly return at least one result (if they locally evaluated q) with a ranking high enough to make it into the global top-k result set (in contrast to S_2, S_7, and S_8, which have upper bounds on their result rankings that are too low to appear in S_L's local top-k). With the sites guaranteed to be false-positives not selected (i.e., the sites outside the boundary), S_L solves the LP problem as formulated in Sec. 3.1 (Line 7). C denotes the set of constraints from Eq. 2 and Eq. 3. For readability, Algorithm 1 does not reflect a possible new value for $|S|$ after eliminating false-positive sites in the previous step (e.g., the call to BoundRankings reduces $|S|$ from 8 to 5 implicitly). In our formulation, the simple solution is to append constraints of $x_{ij} = 0$ to C for each excluded site (e.g., in Fig. 2, $x_{i2} = x_{i7} = x_{i8} = 0$).

Upon completion of LPSolve, $x_{i1} = x_{i3} = x_{i4} = x_{i6} = 1$, which indicates the sites that might contribute results to the global top-k (i.e., the enclosed sites). The subset (i.e., the shaded, enclosed sites) contributes the maximum result ranking upper bounds (i.e., might have at least one document within the top-k ranking) for a budget-restricted energy cost. The global result set R is initialized in Line 8. In Lines 9–14, the query q is forwarded (Line 11) to each member of the subset of plausible non-local sites (Line 11). The results returned to S_L, denoted $\hat{R}$, are then merged into R (Line 12). Finally, in Line 15, the top-k results in R are delivered to the query originator. The remaining $q \in Q$ are processed similarly.

Note that an identical query processed at two different times, say t_i and t_j in Fig. 2, may select a different subset of sites to forward the query to due to spatial and temporal variations in prices. For example, as illustrated in Fig. 2, if the underlying thresholding technique is oblivious to energy prices and documents indexed by the non-local sites do not change, the same subset of non-local sites will be selected for forwarding, i.e., enclosed in the blob-like boundary between time t_i and t_j for the same query. However, the subset selected by RESQ can vary depending on the electricity prices at the sites, e.g., from S_1, S_3, S_4, and S_6 at t_i to S_1, S_4, S_5, and S_6 at t_j. The particular subset depends on the forwarding solution to LP problem for the to-be-evaluated query.

Table 2: A summary of the notation used to describe the RESQ query forwarding model and the distributed algorithm implemented.

Symbol	Explanation
S	set of m search sites
S_j	jth search site
$\hat{S}$	"local" site
I	full search system index
I_j	partial index at site S_j
T	timeline (continuous)
t	time on T
Q	set of n queries at t
q_i	ith query
$r_j(q_i)$	result ranking bound for q_i over I_j
$c_j(q_i)$	cost to evaluate q_i over I_j
x_{ij}	indicator for q_i evaluated at S_j
α_j	energy units / query complexity ratio at S_j
$p_j(t)$	unit energy price at S_j during t

4. EVALUATION

The evaluation of RESQ focuses on the tradeoff between ranking quality and energy cost. The reason is that RESQ is a hybrid selective query forwarding algorithm. We simulate a non-replicated distributed search system that supports the RESQ forwarding model using Java. Using a combination of publicly-available data and principled experimental design, we make direct comparisons between RESQ and realistic rank- and energy-only baselines. In the subsections that follow, we describe our experimental setup, methodology, and results in the subsections that follow. For reproducibility, the code, data, and raw figure values from our study are available online at `https://github.com/resqforwarding/cikm13`.

4.1 Setup

Underlying our evaluation is a combination of the AOL query log [31], New York Independent System Operator (NYISO) electricity prices [29], and Google data center location-related information [18]. We describe each data source and its scope in our study in the subsections that follows.

4.1.1 Query Log

The AOL query log [31] contains approximately $36,000,000$ query events (queries or subsequent result click-through annotations) from over $650,000$ users. Each query event includes the fields `id`, `query terms`, and `time`. Although controversial, the query log is publicly-available and used widely (e.g., [20], [22], [10], [15], [19], and [1]), which simplifies comparisons to past and future results. Existing research has also investigated how to effectively anonymize such query logs (e.g., [21], [11], and [14]). For privacy, we do not consider the linguistic meaning of the terms, and we do not attempt to deanonymize users.

4.1.2 Electricity Prices

The New York Independent System Operator (NYISO) day-ahead zonal market electricity pricing data [29] contains publicly-available electricity price reports from various New York regions in 15-minute intervals. We select a 1-week report duration starting on November 7, 2011. The date is arbitrary. The report from a particular zone forms an energy price time series consisting of 672 (= 1 week $\times$ 7 days/week $\times$ 24 hours/day $\times$ 4 observations/hour) time-price observations of the form $\langle t, p_j(t)\rangle$ where $\langle t\rangle$ is a 15-minute interval from a 24-hour clock period and $p_j(\cdot)$ is the price function in Sec. 3.1. Previous studies incorporating energy prices have used similar data (e.g., [24]).

4.1.3 Data Center Locations

The physical locations of six real Google data centers [18] are selected for estimating site-to-site latencies and subsequent query response times. The locations are approximately uniformly representative of world regions. One site location is "local" and the remaining five sites locations are "non-local" relative to the local site. The quantity of site locations is consistent with previous studies (e.g., [9] and [24]). Table 3 summarizes the locations, including a possible local and non-local assignment, and site-to-site distances based on the assignment. For example, consider a query submitted to a (local) search site in Berkeley County, South Carolina, USA. The great circle distance for forwarding the query to a (non-local) data center in Hamina, Finland is approximately 7716 km. As elaborated on in the next section, the $\mathcal{N}(\cdot,\cdot)$ notation indicates that the example distance is a mean for a distribution of distances employed in the latency construction.

Table 3: The data center locations employed in the evaluation of RESQ. Geographic distributions of distances between the sites are provided for a sample local and non-local assignment.

Location		Distance
Local	Berkeley, SC, USA	$\mathcal{N}(0,0)$
Non-Local	Quilicura, Chile	$\mathcal{N}(7423,1229)$
	St. Ghislain, Belgium	$\mathcal{N}(6840,247)$
	Hamina, Finland	$\mathcal{N}(7716,822)$
	Singapore	$\mathcal{N}(16166,37)$
	Changhua City, Taiwan	$\mathcal{N}(13313,269)$

4.2 Methodology

Although we employ widely-used publicly-available data, our evaluation still relies on simulation. Therefore, we give careful consideration to the overall simulation complexity and the intuition behind the underlying probabilistic structures [35]. In the subsections that follow, we describe the energy prices, latency values, ranking bounds, evaluation configurations, and training and testing processes.

4.2.1 Energy Model

Each of the local and non-local sites are uniformly randomly assigned an energy price time series consisting of observations $\langle t, p_j(t)\rangle$. The series are cyclic shifts of an energy price report (see Sec. 4.1.2) also selected uniformly at random. Intuitively, the shifts capture the site-to-site spatial variation in energy prices. The corresponding 24-hour time stamp of a query is used to assign an electricity price. Specifically, a 24-hour duration in the site's week-long price is uniformly randomly selected, i.e., $p=\frac{1}{7}$, and the query time stamp `time` is used to selected the 15-minute block $\langle t\rangle$ of the duration that the time is contained in. The price during the corresponding block represents the current energy price at site S_j at time t and is denoted $p_j(t)$.

4.2.2 Latency Model

The latency $\hat{L_j}$ between between a local site $\hat{S}$ and non-local site S_j is modeled as a normally distributed random variable. The basic idea is to use a combination of site-to-site geographic distance, the speed of light, and a couple of intuitive network adjustments to estimate latency values. Formally,

$$\hat{L_j} \sim \mathcal{N}(\mu_j, \sigma_j) = y\left(\frac{d\times 1000}{c}\right), \qquad (5)$$

where $d \in \hat{D_j}$, a normally distributed random variable with mean μ_j, the great circle (or "bird flight") distance between $\hat{S}$ and S_j, and standard deviation σ_j, the length of the diagonal formed by treating (the area of) the geographic unit (e.g., a state) containing the location as a square. The intuition is that network connections, even high-speed site-to-site ones, do not maintain static, straight-line source to destination paths, including modest susceptibility to physical or virtual path changes (e.g., non-AS router changes). Note that the bird-flight distance between the sites is measured using geographic locations assigned uniformly randomly to each site from Table 3. Given a distance $d \in \hat{D}_j$, we obtain a latency value by dividing by a constant $c = 200000$ km/s, which is converted to milliseconds (ms), and defined to approximate the speed of light through fiber [32]. However, as pointed out in [9], despite geographic distance-based latency computation being reasonable, other delays (e.g., due to queuing) are not captured. Therefore, we apply the regression-based latency mapping $y = 8.239 + 1.983x$ derived and validated experimentally by Cambazoglu *et al.* [9] to obtain the final latency value for forwarding at time t. Note that a linear function of a normally distributed random variable is also normal.

4.2.3 Ranking Bounds

When a local site processes a query, the result ranking upper bounds are computed as standardized term-length-weighted average frequency counts. Formally, when `query terms` contains k terms $t_1, t_2, \ldots, t_k$, we compute the rank upper bound r_j for results at site S_j as

$$\sum_{i=1}^{k} w_i \times f_j(t_i), \tag{6}$$

where w_i is the average length of term t_i (denoted $|t_i|$) with respect to the lengths of the other terms in the query,

$$w_i = \frac{|t_i|}{\sum_{i=1}^{k} |t_i|}, \tag{7}$$

and $f_j(\cdot)$ is a function that returns the frequency count of a term at S_j. Lastly, the bound is standardized as

$$r_j = \frac{r_j}{\sum_{i=1}^{k} r_i}. \tag{8}$$

Although other term-based approaches are possible [34] [25], the method is simple, has similar intuition, and relates well with site index information. It also adequately maintains the site variability mentioned in Sec. 4.2.4.

4.2.4 Configurations

Each experiment consists of multiple configurations. A configuration consists of a diversity level and an energy budget. The diversity level (denoted L) represents variability in the upper bounds on the result rankings among sites. The intuition behind index diversity is grounded in the localized nature of queries and subsequent retrieved documents. We apply diversity probabilistically over each $\langle f_j(t)\rangle$ in the accumulated $\langle t, f_j(t)\rangle$ pairs at each site (obtained via a process elaborated on in Sec. 4.2.5). Specifically, for a given diversity level L, we let $X_j^i \sim \text{Bernoulli}(1/2)$ be a "coin toss" for the i^{th} $\langle t, f_j(t)\rangle$ pair at S_j. We derive a new frequency $f_j{}'(t)$ for the pair as follows:

$$f_j{}'(t) = f_j(t) \times \begin{cases} 1 + \frac{L}{10} & \text{if } X_j^i = 1, \\ 1 - \frac{L}{10} & \text{otherwise,} \end{cases} \tag{9}$$

and resulting in $\langle t, f_j{}'(t)\rangle$. Note that the diversity levels produce result rankings that generate 10 average standard deviations over the approximately $1,000,000$ queries in each testing set with a Pearson correlation coefficient of almost 1 (i.e., the diversity effectively captures variability in result ranking upper bounds). For an energy budget (denoted μ), the corresponding model term is C in Eq. 2. The budget values are the first five multiples of the average weekly energy price, referred to as $1\mu, \ldots, 5\mu$ in Sec. 4.3.

4.2.5 Training and Testing

As mentioned in Sec. 4.1.3, our experiments employ a total of six sites (or data centers): one local site and five non-local sites. The five non-local sites are trained using $\langle t, f_j(t)\rangle$ pairs accumulated from approximately $18,000,000$ query events, i.e., about half of the query log, distributed evenly among the sites. The query events are preprocessed using a standard approach of lowercase conversion, stemming, and stopword removal. Experiments are conducted using testing data obtained by randomly sampling batches of $1,000,000$ queries from the remaining queries, i.e., the queries that are not used for training. The queries are of the form `<query terms, time>`. We preprocess the `query terms` in the queries using the same standard approach that we applied to the training data. We evaluate the L and μ configurations using 10 testing sets for each combination. The results are averaged, and 95% confidence intervals are computed.[2] Results over all configurations are reported.

4.3 Results

Using the data setup outlined in Sec. 4.1 and the experiment methodology described in Sec. 4.2, we evaluate a distributed search system that implements the RESQ forwarding model. In the subsections that follow, we briefly describe the baseline approaches, performance metrics of interest, and then illustrate and interpret results of direct comparisons between RESQ and the baselines in terms of ranking quality, energy costs incurred, query response times, and the extent of non-local result resilience.

4.3.1 Baselines

Recall that RESQ is a hybrid selective query forwarding model that considers both ranking quality and energy cost. Therefore, we consider the relative performance of RESQ to ranking- and energy-only baseline selective query forwarding algorithms. The first, denoted:

- B_{rank}, represents the ranking-only approach in [9]. The basic idea is to select the subsets of site that guarantee result quality identical to a search system using a fully-replicated index (i.e., not partitioned). The second, denoted:

- B_{energy}, is a heuristic that greedily forwards to non-local sites in increasing order of energy price until a cost budget is exceeded. The underlying intuition is that the quantity of non-local sites contacted is maximized for a given budget.

Note that although we are partially motivated by [24] (see Sec. 2), the approach is not an appropriate baseline in our study because the indexes are replicated fully, resulting in any data center providing equally good scoring documents.

4.3.2 Performance Metrics

An analysis of selective query forwarding algorithms such as RESQ must investigate any relative difference in the ranking quality obtained in the context of baseline approaches.

[2] The error bars did not provide substantial insight and their removal increased illustration clarity. Therefore, we will not discuss their presence in the remaining sections.

On the other hand, as energy costs are a constraint within our problem scope, we're also interested in the relative costs of the different forwarding models. Intuitively, a good selective query forwarding algorithm will balance high quality results with low energy expenses. We formalize the dual goals of result quality and energy cost savings as Δ_ρ, which represents the difference between the ratio of ranking quality upper bounds (or "guarantee factor") and the ratio of cumulative energy cost. The ranking quality and energy cost ratios are denoted ρ_r and ρ_e, respectively. In the context of the ranking-only approach B_{rank}, we let

$$\Delta_\rho = \rho_r - \rho_e, \tag{10}$$

$$\rho_r = \frac{r\,(RESQ)}{r\,(B_{rank})} \text{ and } \rho_c = \frac{c\,(RESQ)}{c\,(B_{rank})}, \tag{11}$$

with the functions r and c as notational conveniences that refer to the cumulative result rankings and energy costs in Eq. 1 and Eq. 2, respectively, for RESQ or the baseline B_{rank}. The definitions are similar for B_{energy}. Intuitively, large positive values of Δ_ρ indicate preservation of ranking quality while reducing energy costs. Indeed, a value of $\rho_r = 1$ indicates that result rankings from RESQ are equivalent to B_{rank}. On the other hand, small values of ρ_e indicate low energy costs relative to B_{rank}.

Also of interest to RESQ is the time from when a query is issued until results are returned or the "query response time." Indeed, while latency from user to local site forwarding is consistent with centralized and non-forwarding approaches, selective query forwarding algorithms such as RESQ also potentially forward queries between sites distributed globally. Therefore, we examine the site-to-site latency of a local site forwarding a query and obtaining results from non-local sites. Based on the formulation in Sec. 3.1, the query response time for a query q_i, denoted $T(q_i)$, is a function of the set of forwarding decisions $\{x_{ij}\}_{j=1}^{m}$ obtained in Line 7 of Algorithm 1 in Sec.3.2 and the site-to-site latency between a local site $\hat{S}$ and non-local site S_j, denoted $\hat{l}_j$, i.e., a value taken by $\hat{L}_j$ via the process discussed in Sec. 4.2.2. Formally,

$$T(q_i) = \max_{1 \le j \le m} \{2 \times \hat{l}_j \times x_{ij} + t_j(q_i)\}, \tag{12}$$

where $t_j(q_i)$ is the processing time for q_i over index I_j at non-local site S_j. The response time is a sum of the longest roundtrip site-latency (among forwarded-to sites) and the time to evaluate the query at the site. Intuitively, the longest-latency forwarded-site is a bottleneck compared to the times of the other sites which already completed the forwarding.

4.3.3 B_{rank} and RESQ

The performance of RESQ relative to ranking-only forwarding baseline B_{rank} is illustrated in Fig. 3. The horizontal axis indicates the diversity in partial indexes, as described in Sec. 4.2.4. The vertical axis reports Δ_ρ, our combined metric for measuring the difference between the ratio of ranking quality upper bounds and the ratio of cumulative energy costs.

The results demonstrate that RESQ maintains favorable percentages of the baseline ranking guarantee while reducing energy costs substantially over a range of index diversity levels and energy cost budgets. For example, with a budget of 2μ, i.e., twice the average weekly energy price, RESQ achieves 76% of the original ranking guarantee at 54% of the energy cost. As shown in the upper right corner of Fig. 3, one of the better tradeoffs occurs when the diversity level is $L = 9$ and the budget is 3μ, i.e., about 84% of the original ranking guarantee with a 47% savings in energy costs.

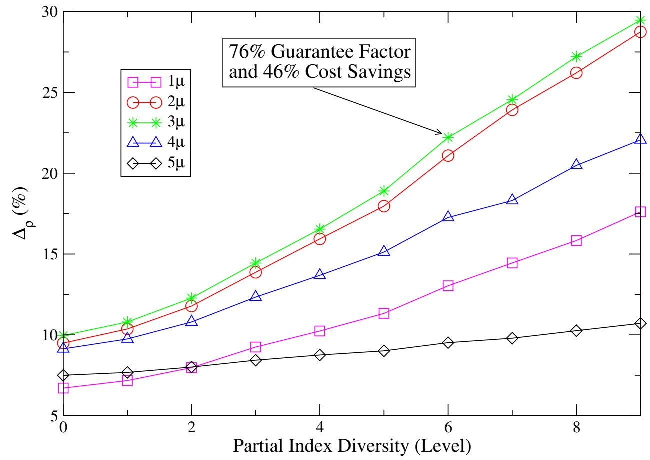


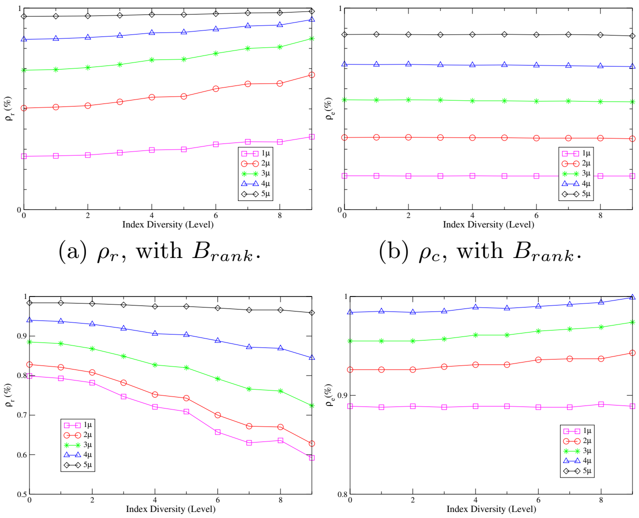
Figure 3: Average performance curves for RESQ and baseline B_{rank} over all testing sets (e.g., 10 trials) and configuration settings (e.g., L and μ).

We examine RESQ's underlying result ranking and energy cost ratios in Fig. 4(a) and Fig. 4(b), respectively. The ρ_r values using the baseline B_{rank} are presented across all diversity levels L and energy budgets μ, i.e, the values under which all of the ranking quality and energy cost tradeoff curves were obtained. For example, the ρ_r curve for B_{rank} in Fig. 4(a) and the ρ_c curves for B_{rank} in Fig. 4(b), both when $\mu = 4$, correspond to the mid-performing curve 4μ in Fig. 3. The steady increase in proportion of the baseline ranking result thresholds is evident across all settings of L and μ, even with relatively constant intra-budget ρ_c values, which we discuss in Sec. 4.3.4.

4.3.4 B_{energy} and RESQ

When compared to B_{energy}, the energy consumption for a forwarding algorithm such as RESQ should be as close to the greedy baseline as possible. The intuition is that energy cost constraint may be considered to be tuned to a desired cost. Under such a scenario, the best case ratio occurs when $\rho_c = 1$. On the other hand, an increase in ranking quality over the baseline is still desirable. Therefore, we let $\Delta_\rho = \rho_c - \rho_r$, with $\rho_c = \frac{c(B_{energy})}{c(RESQ)}$ and $\rho_r = \frac{r(B_{energy})}{r(RESQ)}$, i.e., we invert Eq. (10) and Eq. (11) in Sec. 4.3.2. As demonstrated by the performance curves in Fig. 5, RESQ leverages relatively small differences in energy costs for substantially increased improvement in ranking quality. For example, with a medium level of index variability of $L = 6$ and an energy budget of 3μ, RESQ gets 7% closer to the desired cost while increasing the result quality by 14%. The improvement is consistent across the parameter settings.

We also examine RESQ's underlying result ranking and energy cost ratios in Fig. 4(c) and Fig. 4(d), respectively. Recall that ρ_r is inverted for B_{energy}, which makes the decreasing values in the budget curves indicated improving performance. Although important to the overall performance illustrated in Fig. 3 and Fig. 5, the ρ_c values in Fig.



(a) ρ_r, with B_{rank}.

(b) ρ_c, with B_{rank}.

(c) ρ_r, with B_{energy}.

(d) ρ_c, with B_{energy}.

Figure 4: Relative performance of RESQ to: 4(a), B_{rank} with respect to result rankings; 4(b), B_{rank} with respect to energy cost; 4(c), B_{energy} with respect to result rankings; and 4(d), B_{energy} with respect to energy costs. Results are averages over all testing sets (e.g., 10 trials) and configuration settings (e.g., L and μ).

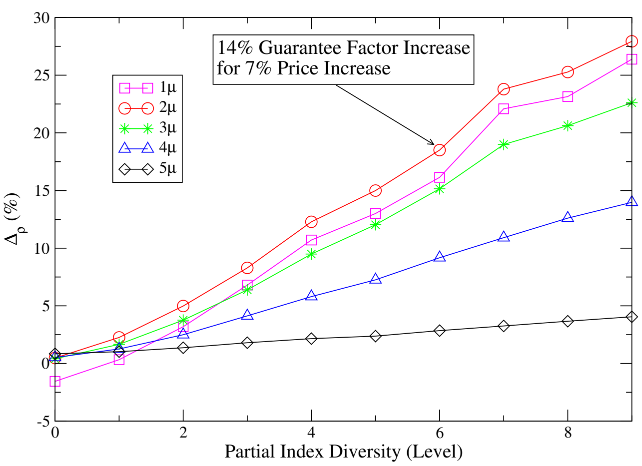


Figure 5: Average performance curves for RESQ and baseline B_{energy} over all testing sets (e.g., 10 trials) and configuration settings (e.g., L and μ).

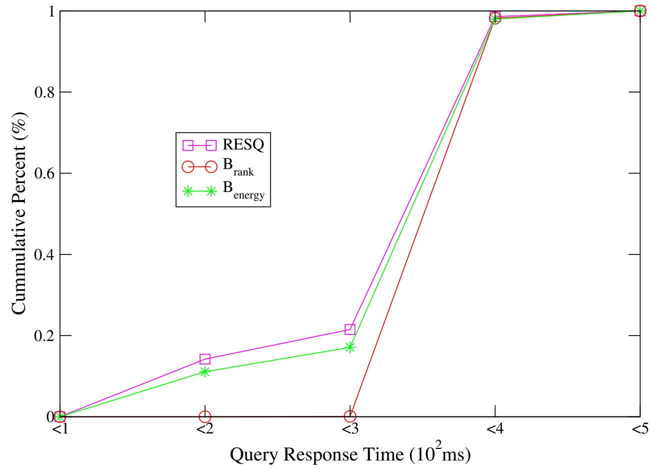


Figure 6: Query response times CDFs for B_{rank}, B_{energy}, and RESQ. The energy budget is 4μ and the diversity level is $L = 5$, which represents a configuration that produces an average result.

4(b) and Fig. 4(d) did not show substantial intra-budget variation. The reason is that the coarse granularity of energy budgets relative to site energy prices made sites similarly expensive under a particular budget. In this respect, the results obtained are pessimistic, as the hybrid nature of RESQ allows for maximal savings under nuanced variance in energy prices or budgets.

4.3.5 Query Response Time

Sequential query forwarding algorithms must deliver high quality results and cost savings in spite of communication latency across potentially large geographic distances (e.g., between the locations in Table 3). Therefore, we investigate the query response times obtained by RESQ and the baselines B_{rank} and B_{energy}. That is, we measure the time between when a local site initially forwards a query to when the non-local documents are returned to the forwarding site. The response time results for each forwarding algorithm are illustrated with a cumulative distribution function (CDF) in Fig. 6.

For the CDFs shown, the fraction of queries that are answered under a particular response time in milliseconds for B_{rank}, B_{energy}, and the proposed algorithm RESQ is computed using Eq. (12) in Sec. 4.3.2. Averages are computed over a run of experiments and reported in 100 ms intervals. The configuration settings of L=5 and 4μ were selected because of the medium level of results the configuration produced in Fig. 3. RESQ forwards queries in shorter times than each of the baselines, with B_{rank} taking the longest amount of the time. The close response time distributions of B_{energy} and RESQ are due to neither algorithm explicitly making forwarding decisions based on latency. We note that the RESQ forwarding model in Sec. 3.1 does not preclude the addition of latency based constraints. However, we do not explore such extensions in the current study. The percentage of queries that take less time correspond to situations where a longer-latency site was not contacted according to a portion of the LP solution contained in Eq. (4). The intuition is that B_{rank}, for example, forwards queries to all sites that may contribute top-ranked results, including sites

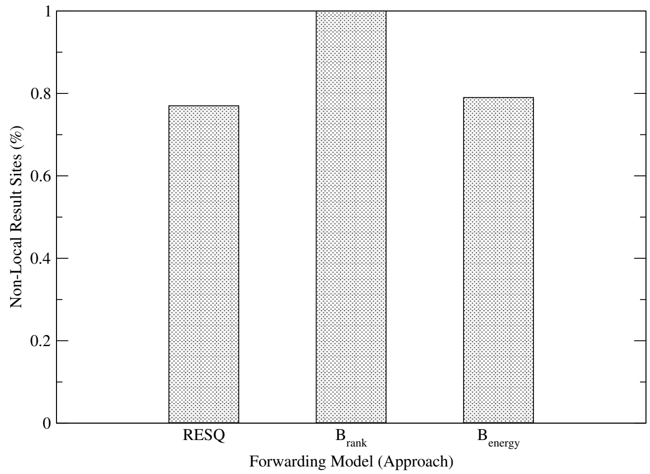


Figure 7: Average fraction of non-local sites selected from the underlying thresholding scheme that receive queries using B_{rank}, B_{energy}, and RESQ. The energy budget is 4μ and the diversity level is $L = 5$, which represents a configuration that produces an average result.

that may incur a long latency. RESQ, on the other hand, answers about 21% of its queries in less time while maintaining substantial ranking guarantees and saving energy costs (e.g., the Δ_ρ values in Fig. 3).

4.3.6 Non-Local Result Site Distribution

A characterization of the number of non-local sites employed by selective query forwarding algorithm also provides insight. Indeed, with more sites contacted, there is more of a chance for efficiency gains from query forwarding to be diminished (e.g., consider a degenerate case where all non-local sites are considered to possibly have relevant documents). Although some of the related efforts report the fraction of non-local sites their approaches contact (e.g., [9] and [24]), the fraction of result sites selected by the underlying thresholding technique (or "result site distribution") is more relevant. The reason is that RESQ forwards to a subset of the sites that the base thresholding technique selects (i.e., the algorithm never forwards to a site that does not have the possibility of returning a top-k result). The average percentage of the non-local sites that might have results scoring well enough to warrant the locally-received query being forwarded to them using B_{rank}, B_{energy}, and RESQ is illustrated in Fig. 7. Recall that as B_{rank} selects all of the thresholded sites, its fraction is always 1. Compared to B_{energy}, RESQ is subject to identical budgets and coarse granularity in prices. Hence, over a small number of sites, there is not a substantial difference in the fraction of the non-local sites contacted. On the other hand, compared to B_{rank}, RESQ selects a (possibly proper) subset of the sites to forward queries to. On average, RESQ makes about 23% fewer forwards, while maintaining a substantially higher percentage of the result ranking upperbounds, and less relative energy costs, as illustrated in Fig. 3.

4.3.7 Remark

We note that ranking quality thresholding techniques cannot guarantee zero false-positive sites without all non-local sites evaluating a query (which obviates the need for selective query forwarding). Indeed, it is possible that the locally-aggregated results contain only local-results for the global top-k. For example, as mentioned in Sec. 3.2, the non-local sites selected for forwarding might return results that all score lower than documents at the local site. Therefore, the ratio of the result rankings is more relevant to RESQ than the false-positives associated with the underlying thresholding technique (e.g., [9]). We also point out that a decrease in ranking guarantee is not necessarily equivalent to a downgrade in ranking quality. Indeed, the results of such systems are a function of retrieved document scores, with properties of the underlying bounds determining the possible quality of results aggregated at a local site.

5. CONCLUSION

We introduced RESQ, a novel hybrid rank-energy selective forwarding model that considers ranking quality and energy costs simultaneously in non-replicated distributed search systems. Experiments with the AOL query log, NYISO energy price series, and real Google data center locations demonstrate that RESQ forwarding achieves favorable tradeoffs between the possibility of returning high quality query results and the costs due to temporal and spatial variations in energy prices.

An interesting future direction for our study is to investigate rank-energy approaches to search engine caching. For example, consideration of the spatial and temporal variance in energy prices that RESQ exploits may lead to increased cost savings for cache eviction algorithms (e.g., [16] and [7]). Similarly, adaptive selection of underlying site retrieval strategies (e.g., based on query efficiency prediction [5]) may also help RESQ process the queued query volume at a local site within particularly low- or high-price time durations. While the design of RESQ does not necessarily preclude the adoption of such existing approaches, a fresh look at the problems in the context of rank- and energy-awareness for distributed search systems may yield useful insights.

6. ACKNOWLEDGMENTS

The authors thank Xiao Qin of Worcester Polytechnic Institute for early design discussions, and the three anonymous reviewers for detailed comments and helpful suggestions.

7. REFERENCES

[1] I. S. Altingövde, R. Blanco, B. B. Cambazoglu, R. Ozcan, E. Sarigil, and Ö. Ulusoy. Characterizing web search queries that match very few or no results. In *ACM CIKM '12*.

[2] R. Baeza-Yates, A. Gionis, F. Junqueira, V. Plachouras, and L. Telloli. On the feasibility of multi-site web search engines. In *ACM CIKM '09*.

[3] C. Baumgarten. A probabilistic solution to the selection and fusion problem in distributed information retrieval. In *ACM SIGIR '99*.

[4] R. Blanco, B. B. Cambazoglu, F. P. Junqueira, I. Kelly, and V. Leroy. Assigning documents to master sites in distributed search. In *ACM CIKM '11*.

[5] D. Broccolo, C. Macdonald, S. Orlando, I. Ounis, R. Perego, F. Silvestri, and N. Tonellotto.

Load-sensitive selective pruning for distributed search. In *ACM CIKM '13*.

[6] B. Cahoon and K. S. McKinley. Performance evaluation of a distributed architecture for information retrieval. In *ACM SIGIR '96*.

[7] B. B. Cambazoglu, F. P. Junqueira, V. Plachouras, S. Banachowski, B. Cui, S. Lim, and B. Bridge. A refreshing perspective of search engine caching. In *WWW '10*.

[8] B. B. Cambazoglu, V. Plachouras, and R. Baeza-Yates. Quantifying performance and quality gains in distributed web search engines. In *ACM SIGIR '09*.

[9] B. B. Cambazoglu, E. Varol, E. Kayaaslan, C. Aykanat, and R. Baeza-Yates. Query forwarding in geographically distributed search engines. In *ACM SIGIR '10*.

[10] M. J. Carman, F. Crestani, M. Harvey, and M. Baillie. Towards query log based personalization using topic models. In *ACM CIKM '10*.

[11] G. Chen, H. Bai, L. Shou, K. Chen, and Y. Gao. UPS: Efficient privacy protection in personalized web search. In *ACM SIGIR '11*.

[12] comScore. comscore releases march 2012 u.s. search engine rankings. `http://www.comscore.com/Insights/Press_Releases/2012/4/comScore_Releases_March_2012_U.S._Search_Engine_Rankings`.

[13] G. Das, D. Gunopulos, N. Koudas, and D. Tsirogiannis. Answering top-k queries using views. In *VLDB '06*.

[14] H. Feild, J. Allan, and J. Glatt. Crowdlogging: Distributed, private, and anonymous search logging. In *ACM SIGIR '11*.

[15] M. Fontoura, M. Gurevich, V. Josifovski, and S. Vassilvitskii. Efficiently encoding term co-occurrences in inverted indexes. In *ACM CIKM '11*.

[16] Q. Gan and T. Suel. Improved techniques for result caching in web search engines. In *WWW '09*.

[17] D. Goodwin. Spotlight Keynote with Matt Cutts. `http://searchenginewatch.com/article/2199092/Spotlight-Keynote-With-Matt-Cutts-SESSF`.

[18] Google. Data center locations. `http://www.google.com/about/datacenters/inside/locations`.

[19] S. Gurajada and S. K. P. Index tuning for query-log based on-line index maintenance. In *ACM CIKM '11*.

[20] S. Gurajada and S. K. P. On-line index maintenance using horizontal partitioning. In *ACM CIKM '09*.

[21] Y. Hong, X. He, J. Vaidya, N. Adam, and V. Atluri. Effective anonymization of query logs. In *ACM CIKM '09*.

[22] J. Huang and E. N. Efthimiadis. Analyzing and evaluating query reformulation strategies in web search logs. In *ACM CIKM '09*.

[23] F. P. Junqueira, V. Leroy, and M. Morel. Reactive index replication for distributed search engines. In *ACM SIGIR '12*.

[24] E. Kayaaslan, B. B. Cambazoglu, R. Blanco, F. P. Junqueira, and C. Aykanat. Energy-price-driven query processing in multi-center web search engines. In *ACM SIGIR '11*.

[25] Y. Ko. A study of term weighting schemes using class information for text classification. In *ACM SIGIR '12*.

[26] A. Kulkarni and J. Callan. Document allocation policies for selective searching of distributed indexes. In *ACM CIKM '10*.

[27] A. Kulkarni, A. S. Tigelaar, D. Hiemstra, and J. Callan. Shard ranking and cutoff estimation for topically partitioned collections. In *ACM CIKM '12*.

[28] A. Moffat, W. Webber, and J. Zobel. Load balancing for term-distributed parallel retrieval. In *ACM SIGIR '06*.

[29] New York Independent System Operator (NYISO). Pricing data. `http://www.nyiso.com/public/markets_operations/market_data/pricing_data/index.jsp`.

[30] J. Niccolai. Swedish data center saves $1 million a year using seawater for cooling. `http://www.pcworld.com/article/2038934/swedish-data-center-saves-1-million-a-year-using-seawater-for-cooling.html`.

[31] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *ACM InfoScale '06*.

[32] B. Quigley. Speed of light in fiber–the first building block of a low-latency trading infrastructure. `http://blog.advaoptical.com/speed-light-fiber-first-building-block-low-latency-trading-infrastructure`.

[33] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting the electric bill for internet-scale systems. In *ACM SIGCOMM '09*.

[34] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.

[35] J. Tague, M. Nelson, and H. Wu. Problems in the simulation of bibliographic retrieval systems. In *ACM SIGIR '80*.

[36] A. Teymorian, X. Qin, and O. Frieder. Resq: rank-energy selective query forwarding for distributed search systems. In *ACM CIKM '12*.

[37] U.S. Energy Information Administration (EIA). Electricity prices, selected countries. `http://www.eia.gov/countries/prices/electricity_industry.cfm`.

[38] YouTube. Press room statistics. `http://www.youtube.com/t/press_statistics`.

[39] Q. Zhang, Y. Wu, Z. Ding, and X. Huang. Learning hash codes for efficient content reuse detection. In *ACM SIGIR '12*.