Respect My Authority! HITS Without Hyperlinks, Utilizing Cluster-Based Language Models

Oren Kurland and Lillian Lee
Department of Computer Science
Cornell University
Ithaca, NY 14853-7501
kurland@cs.cornell.edu, llee@cs.cornell.edu

ABSTRACT

We present an approach to improving the precision of an initial document ranking wherein we utilize cluster information within a graph-based framework. The main idea is to perform re-ranking based on *centrality* within bipartite graphs of documents (on one side) and clusters (on the other side), on the premise that these are mutually reinforcing entities. Links between entities are created via consideration of language models induced from them.

We find that our cluster-document graphs give rise to much better retrieval performance than previously proposed document-only graphs do. For example, authority-based re-ranking of documents via a HITS-style cluster-based approach outperforms a previously-proposed PageRank-inspired algorithm applied to solely-document graphs. Moreover, we also show that computing authority scores for clusters constitutes an effective method for identifying clusters containing a large percentage of relevant documents.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval models

General Terms: Algorithms, Experimentation

Keywords: bipartite graph, clusters, language modeling, HITS, hubs, authorities, PageRank, high-accuracy retrieval, graph-based retrieval, structural re-ranking, cluster-based language models

1. INTRODUCTION

To improve the precision of retrieval output, especially within the very few (e.g, 5 or 10) highest-ranked documents that are returned, a number of researchers [36, 13, 16, 7, 22, 34, 25, 1, 18, 9] have considered a *structural re-ranking* strategy. The idea is to re-rank the top N documents that some initial search engine produces, where the re-ordering utilizes information about inter-document relationships within that set. Promising results have been previously obtained by using document *centrality* within the initially retrieved list to perform structural re-ranking, on the premise that if the quality of this list is reasonable to begin with, then the documents that are most related to most of the documents on the list are likely to be the most relevant ones. In particular, in our prior work [18] we adapted *PageRank* [3] — which, due to the success of Google, is surely the most well-established algorithm for defining and computing centrality within a directed graph — to the task of re-ranking non-hyperlinked document sets.

The arguably most well-known alternative to PageRank is Kleinberg's *HITS* algorithm [16]. The major conceptual way in which HITS differs from PageRank is that it defines two different types of central items: each node is assigned both a *hub* and an *authority* score as opposed to a single PageRank score. In the Web setting, in which HITS was originally proposed, good hubs correspond roughly to high-quality resource lists or collections of pointers, whereas good authorities correspond to the high-quality resources themselves; thus, distinguishing between two differing but interdependent types of Webpages is quite appropriate. Our previous study [18] applied HITS to non-Web documents. We found that its performance was comparable to or better than that of algorithms that do not involve structural re-ranking; however, HITS was not as effective as PageRank [18].

Do these results imply that PageRank is better than HITS for structural re-ranking of non-Web documents? Not necessarily, because there may exist graph-construction methods that are more suitable for HITS. Note that the only entities considered in our previous study were documents. If we could introduce entities distinct from documents but enjoying a mutually reinforcing relationship with them, then we might better satisfy the spirit of the hubs-versus-authorities distinction, and thus derive stronger results utilizing HITS.

A crucial insight of the present paper is that document *clusters* appear extremely well-suited to play this complementary role. The intuition is that: (a) given those clusters that are "most representative" of the user's information need, the documents within those clusters are likely to be relevant; and (b) the "most representative" clusters should be those that contain many relevant documents. This apparently circular reasoning is strongly reminiscent of the interrelated hubs and authorities concepts underlying HITS.

Also, clusters have long been considered a promising source of information. The well-known *cluster hypothesis* [35] encapsulates the intuition that clusters can reveal groups of relevant documents; in practice, the potential utility of clustering for this purpose has been demonstrated for both the case wherein clusters were created in a query-independent fashion [14, 4] and the re-ranking setting [13, 22, 34].

SIGIR'06, August 6–11, 2006, Seattle, Washington, USA.

In this paper, we show through an array of experiments that consideration of the mutual reinforcement of clusters and documents in determining centrality can lead to highly effective algorithms for re-ranking an initially retrieved list. Specifically, our experimental results show that the centrality-induction methods that we previously studied solely in the context of document-only graphs [18] result in much better re-ranking performance if implemented over bipartite graphs of documents (on one side) and clusters (on the other side). For example, ranking *documents* by their "authoritativeness" as computed by HITS upon these cluster-document graphs yields better performance than that of a previously proposed PageRank implementation applied to document-only graphs. Interestingly, we also find that *cluster* authority scores can be used to identify clusters containing a large percentage of relevant documents.

2. ALGORITHMS FOR RE-RANKING

Since we are focused on the structural re-ranking paradigm, our algorithms are applied not to the entire corpus, but to a subset $\mathcal{D}_{\text{init}}^{N,q}$ (henceforth $\mathcal{D}_{\text{init}}$), defined as the top N documents retrieved in response to the query q by a given initial retrieval engine. Some of our algorithms also take into account a set $Cl(\mathcal{D}_{\text{init}})$ of *clusters* of the documents in $\mathcal{D}_{\text{init}}$. We use $\mathcal{S}_{\text{init}}$ to refer generically to whichever set of entities — either $\mathcal{D}_{\text{init}}$ or $\mathcal{D}_{\text{init}} \cup Cl(\mathcal{D}_{\text{init}})$ — is used by a given algorithm.

The basic idea behind the algorithms we consider is to determine centrality within a *relevance-flow graph*, defined as a directed graph with non-negative weights on the edges in which

- the nodes are the elements of $\mathcal{S}_{\text{init}}$, and

- the weight on an edge between node u and v is based on the strength of evidence for v's relevance that would follow from an assertion that u is relevant.

By construction, then, any measure of the centrality of $s \in \mathcal{S}_{\text{init}}$ should measure the accumulation of evidence for its relevance according to the set of interconnections among the entities in $\mathcal{S}_{\text{init}}$. Such information can then optionally be subjected to additional processing, such as integration with information on each item's similarity to the query, to produce a final re-ranking of $\mathcal{D}_{\text{init}}$.

Conventions regarding graphs. The types of relevance-flow graphs we consider can all be represented as weighted directed graphs of the form (V, wt), where V is a finite non-empty set of nodes and $wt : V \times V \to [0, \infty)$ is a non-negative edge-weight function. Note that thus our graphs technically have edges between all ordered pairs of nodes (self-loops included); however, edges with zero edge-weight are conceptually equivalent to missing edges. For clarity, we write $wt(u \to v)$ instead of $wt(u, v)$.

2.1 Hubs, authorities, and the HITS algorithm

The HITS algorithm for computing centrality can be motivated as follows. Let $G = (V, wt)$ be the input graph, and let v be a node in V. First, suppose we somehow knew the *hub score* $\mathsf{hub}(u)$ of each node $u \in V$, where "hubness" is the extent to which the nodes that u points to are "good" in some sense. Then, v's *authority score*

$$\mathsf{auth}(v) = \sum_{u \in V} wt(u \to v) \cdot \mathsf{hub}(u) \quad (1)$$

would be a natural measure of how "good" v is, since a node that is "strongly" pointed to by high-quality hubs (which, by definition, tend to point to "good" nodes) receives a high score. But where do we get the hub score for a given node u? A natural choice is to use the extent to which u "strongly" points to highly authoritative nodes:

$$\mathsf{hub}(u) = \sum_{v \in V} wt(u \to v) \cdot \mathsf{auth}(v). \quad (2)$$

Clearly, Equations 1 and 2 are mutually recursive. However, the iterative HITS algorithm[1] provably converges to (non-identically-zero, non-negative) score functions hub^* and auth^* that satisfy the above pair of equations.

Figure 1 depicts the "iconic" case in which the input graph G is *one-way bipartite*, that is, V can be partitioned into non-empty sets V_{Left} and V_{Right} such that only edges in $V_{\text{Left}} \times V_{\text{Right}}$ can receive positive weight, and $\forall u \in V_{\text{Left}}$, $\sum_{v \in V_{\text{Right}}} wt(u \to v) > 0$. It is the case that $\mathsf{auth}^*(u) = 0$ for every $u \in V_{\text{Left}}$ and $\mathsf{hub}^*(v) = 0$ for every $v \in V_{\text{Right}}$; in this sense, the left-hand nodes are "pure" hubs and the right-hand nodes are "pure" authorities.

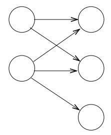

Figure 1: A one-way bipartite graph. We only show positive-weight edges (omitting weight values). According to HITS, the left-hand nodes are (pure) hubs; the right-hand ones are (pure) authorities.

Note that in the end, we need to produce a *single* centrality score for each node $n \in V$. For experimental simplicity, we consider only two possibilities in this paper — using $\mathsf{auth}^*(n)$ as the final centrality score, or using $\mathsf{hub}^*(n)$ instead— although combining the hub and authority scores is also an interesting possibility.

2.2 Graph schemata: incorporating clusters

Recall that the fundamental operation in our structural re-ranking paradigm is to compute the centrality of entities (with)in a set $\mathcal{S}_{\text{init}}$. One possibility is to define $\mathcal{S}_{\text{init}}$ as $\mathcal{D}_{\text{init}}$, the documents in the initially retrieved set; we refer generically to any relevance-flow graph induced under this choice as a ***document-to-document*** graph. But note that for non-Web documents, it may not be obvious *a priori* what kinds of documents are hubs and what kinds are authorities.

Alternatively, we can define $\mathcal{S}_{\text{init}}$ as $\mathcal{D}_{\text{init}} \cup Cl(\mathcal{D}_{\text{init}})$, where $Cl(\mathcal{D}_{\text{init}})$ consists of clusters of the documents in $\mathcal{D}_{\text{init}}$. On a purely formal level, doing so allows us to map the hubs/authorities duality discussed above onto the documents/clusters duality, as follows. Recalling our discussion of the "iconic" case of one-way bipartite graphs $G = ((V_{\text{Left}}, V_{\text{Right}}), wt)$, we can create ***document-as-authority*** graphs simply by

[1]Strictly speaking, the algorithm and proof of convergence as originally presented [16] need (trivial) modification to apply to edge-weighted graphs.

choosing $V_{\text{Left}} = Cl(\mathcal{D}_{\text{init}})$ and $V_{\text{Right}} = \mathcal{D}_{\text{init}}$, so that necessarily clusters serve the role of (pure) hubs and documents serve the role of (pure) authorities. Contrariwise,[2] we can create ***document-as-hub*** graphs by setting $V_{\text{Left}} = \mathcal{D}_{\text{init}}$ and $V_{\text{Right}} = Cl(\mathcal{D}_{\text{init}})$.

But the advantages of incorporating cluster-based information are not just formal. The well-known *cluster hypothesis* [35] encapsulates the intuition that clusters can reveal groups of relevant documents; in practice, the potential utility of clustering for this purpose has been demonstrated a number of times, whether the clusters were created in a query-independent fashion [14, 4], or from the initially most-highly-ranked documents for some query [13, 22, 34] (i.e., in the re-ranking setting). Since central clusters are, supposedly, those that accrue the most evidence for relevance, documents that are strongly identified with such clusters should themselves be judged highly relevant.[3] [4] But identifying such clusters is facilitated by knowledge of which documents are most likely to be relevant — exactly the mutual reinforcement property that HITS was designed to leverage.

2.3 Alternative scores: PageRank and influx

We will compare the results of using the HITS algorithm against those derived using PageRank instead. This is a natural comparison because PageRank is the most well-known centrality-induction algorithm utilized for ranking documents, and because in earlier work [18], PageRank performed quite well as a tool for structural re-ranking of non-Web documents, at least when applied to document-to-document graphs.

One can think of PageRank as a version of HITS in which the hub/authority distinction has been collapsed. Thus, writing "PR" for both auth and hub, we *conceptually* have the (single) equation

$$\mathsf{PR}(v) = \sum_{u \in V} wt(u \to v) \cdot \mathsf{PR}(u). \tag{3}$$

However, in practice, we incorporate Brin and Page's smoothing scheme [3] together with a correction for nodes with no positive-weight edges emanating from them [27, 21]:

$$\begin{aligned} \mathsf{PR}(v) &= \sum_{u \in V: out(u) > 0} \left[\frac{(1-\lambda)}{|V|} + \lambda \frac{wt(u \to v)}{out(u)} \right] \cdot \mathsf{PR}(u) \\ &\quad + \sum_{u \in V: out(u) = 0} \frac{1}{|V|} \cdot \mathsf{PR}(u) \end{aligned} \tag{4}$$

where $out(u) \stackrel{def}{=} \sum_{v' \in V} wt(u \to v')$, and $\lambda \in (0, 1)$ is the damping factor.[5]

Equation 4 is recursive, but there are iterative algorithms that provably converge to the unique positive solution PR^* satisfying the sum-normalization constraint $\sum_{v \in V} \mathsf{PR}(v) = 1$ [21]. Moreover, a (non-trivial) *closed-form* — and quite easily computed — solution exists for one-way bipartite graphs:

THEOREM 1. *If $G = (V, wt)$ is one-way bipartite, then*

$$\mathsf{PR}_{bip}(v) \stackrel{def}{=} \sum_{u \in V: out(u) > 0} \frac{wt(u \to v)}{out(u)} \tag{5}$$

is an affine transformation (with respect to positive constants) of, and therefore equivalent for ranking purposes to, the unique positive sum-normalized solution to Equation 4.

(Proof omitted due to space constraints.) Interestingly, this result shows that while one might have thought that clusters and documents would "compete" for PageRank score when placed within the same graph, in our document-as-authority and document-as-hub graphs this is not the case.

Earlier work [18] also considered scoring a node v by its *influx*, $\sum_{u \in V} wt(u \to v)$. This can be viewed as either a non-recursive version of Equation 3, or as an un-normalized analog of Equation 5.

2.4 Algorithms based on centrality scores

Clearly, we can rank documents by their scores as computed by any of the functions introduced above. But when we operate on document-as-authority or document-as-hub graphs, centrality scores for the clusters are also produced. These can be used to derive alternative means for ranking documents. We follow Liu and Croft's approach [25]: first, rank the documents within (or most strongly associated to) each cluster according to the initial retrieval engine's scores; then, derive the final list by concatenating the within-cluster lists in order of decreasing cluster score, discarding repeats. Such an approach would be successful if cluster centrality is strongly correlated with the property of containing a large percentage of relevant documents.

Ranking algorithms. Since we have two possible ranking paradigms, we adopt the following **algorithm naming conventions**. Names consist of a hyphen-separated prefix and suffix. The prefix ("doc" or "clust") indicates whether documents were ranked directly by their centrality scores, or indirectly through the concatenation process outlined above in which it is the *clusters'* centrality scores that were employed. The suffix ("Auth", "Hub", "PR", or "Influx") indicates which score function (auth^*, hub^*, PR^* (or PR_{bip}), or influx) was used to measure centrality. For a given re-ranking algorithm, we indicate the graph upon which it was run in brackets, e.g., "doc-Auth[G]".

3. RELATED WORK

The potential merits of *query-dependent clustering*, that is, clustering the documents retrieved in response to a query, have long been recognized [30, 36, 23, 34, 25], especially in interactive retrieval settings [13, 22, 32]. However, automatically detecting clusters that contain many relevant documents remains a very hard task [36]. Section 5.2 presents re-

[2]In practice, one can simultaneously compute the output of HITS for a given document-as-authority and document-as-hub graph *pair* by "overlaying" the two into a single graph and suitably modifying HITS's normalization scheme.

[3]We say "are strongly identified with", as opposed to "belong to" to allow for overlapping or probabilistic clusters. Indeed, the one-way bipartite graphs we construct are ill-suited to the HITS algorithm if document-to-cluster links are based on membership in disjoint clusters.

[4]This is, in some sense, a type of smoothing: a document might be missing some of the query terms (perhaps due to synonymy), but if it lies within a sector of "document space" containing many relevant documents, it could still be deemed highly relevant. Recent research pursues this smoothing idea at a deeper level [25, 17].

[5]Under the original "random surfer" model, the sum of the transition probabilities out of "no outflow" nodes — which are abundant in one-way bipartite graphs — would be $(1 - \lambda)$, not 1. Conceptually, the role of the second summation in Equation 4 is to set $\lambda = 0$ for these no-outflow nodes.

sults for detecting such clusters using centrality-based cluster ranking.

Recently, there has been a growing body of work on graph-based modeling for different language-processing tasks wherein links are induced by inter-entity textual similarities. Examples include document (re-)ranking [7, 24, 9, 18, 39], text summarization [11, 26], sentence retrieval [28], and document representation [10]. In contrast to our methods, links connect entities of the same type, and clusters of entities are not modeled within the graphs.

While ideas similar to ours by virtue of leveraging the mutual reinforcement of entities of different types, or using bipartite graphs of such entities for clustering (rather than using clusters), are abundant (e.g., [15, 8, 2]), we focus here on exploiting mutual reinforcement in ad hoc retrieval.

Random walks (with early stopping) over bipartite graphs of terms and documents were used for query expansion [20], but in contrast to our work, no stationary solution was sought. A similar "short chain" approach utilizing bipartite graphs of clusters and documents for ranking an entire corpus was recently proposed [19], thereby constituting the work most resembling ours. However, again, a stationary distribution was not sought. Also, *query drift* prevention mechanisms were required to obtain good performance; in our re-ranking setting, we need not employ such mechanisms.

4. EVALUATION FRAMEWORK

Most aspects of the evaluation framework described below are adopted from our previous experiments with non-cluster-based structural re-ranking [18] so as to facilitate direct comparison. Section 4.1 of [18] provides a more detailed justification of the experimental design. The main conceptual changes[6] here are: a slightly larger parameter search-space for the "out-degree" parameter δ (called the "ancestry" parameter α in [18]); and, of course, the incorporation of clusters.

4.1 Graph construction

Relevance flow based on language models (LMs). To estimate the degree to which one item, if considered relevant, can vouch for the relevance of another, we follow our previous work on document-based graphs [18] and utilize $p_d^{[\mu]}(\cdot)$, the unigram Dirichlet-smoothed language model induced from a given document d (μ is the smoothing parameter) [38]. To adapt this estimation scheme to settings involving clusters, we derive the language model $p_c^{[\mu]}(\cdot)$ for a cluster c by treating c as the (large) document formed by concatenating[7] its constituent (or most strongly associated) documents [17, 25, 19].

The relevance-flow measure we use is essentially a directed similarity in language-model space:

$$rflow(x, y) \stackrel{def}{=} \exp\left(-D\left(p_x^{[0]}(\cdot) \;\middle|\middle|\; p_y^{[\mu]}(\cdot)\right)\right), \qquad (6)$$

where D is the Kullback-Leibler divergence. The asymmetry of this measure corresponds nicely to the intuition that relevance flow is not symmetric [18]. Moreover, this function is somewhat insensitive to large length differences between the items in question [18], which is advantageous when both documents and clusters (which we treat as very long documents) are considered.

Previous work [18, 33] makes heavy use of the idea of nearest neighbors in language-model space. It is therefore convenient to introduce the notation $Nbhd(x \mid m, R)$, pronounced "neighborhood", to denote the m items y within the "restriction set" R that have the highest values of $rflow(x, y)$ (we break ties by item ID, assuming that these have been assigned to documents and clusters). Note that the neighborhood of x corresponds to what we previously termed the "top generators" of x [18].

Graphs used in experiments. For a given set $\mathcal{D}_{\text{init}}$ of initially retrieved documents and positive integer δ (an "out-degree" parameter), we consider the following three graphs. Each connects nodes u to the δ other nodes, drawn from some specified set, that u has the highest relevance flow to.

The document-to-document graph **d↔d** has vertex set $\mathcal{D}_{\text{init}}$ and weight function

$$wt^{\text{d}\leftrightarrow\text{d}}(u, v) = \begin{cases} rflow(u, v) & \text{if } v \in Nbhd(u \mid \delta, \mathcal{D}_{\text{init}} - \{u\}), \\ 0 & \text{otherwise.} \end{cases}$$

The document-as-authority graph **c→d** has vertex set $\mathcal{D}_{\text{init}} \cup Cl(\mathcal{D}_{\text{init}})$ and a weight function such that positive-weight edges go only from clusters to documents:

$$wt^{\text{c}\rightarrow\text{d}}(u, v) = \begin{cases} rflow(u, v) & \text{if } u \in Cl(\mathcal{D}_{\text{init}}) \text{ and} \\ & \quad v \in Nbhd(u \mid \delta, \mathcal{D}_{\text{init}}), \\ 0 & \text{otherwise.} \end{cases}$$

The document-as-hub graph **d→c** has vertex set $\mathcal{D}_{\text{init}} \cup Cl(\mathcal{D}_{\text{init}})$ and a weight function such that positive-weight edges go only from documents to clusters:

$$wt^{\text{d}\rightarrow\text{c}}(u, v) = \begin{cases} rflow(u, v) & \text{if } u \in \mathcal{D}_{\text{init}} \text{ and} \\ & \quad v \in Nbhd(u \mid \delta, Cl(\mathcal{D}_{\text{init}})), \\ 0 & \text{otherwise.} \end{cases}$$

Since the latter two graphs are one-way bipartite, Theorem 1 applies to them.

Clustering Method. Clearly, our cluster-based graphs require the construction of clusters of the documents in $\mathcal{D}_{\text{init}}$. Since this set is query-dependent, at least some of the clustering process must occur at retrieval time, mandating the use of extremely efficient algorithms [6, 37]. The approach we adopt is to use overlapping nearest-neighbor clusters, which have formed the basis of effective retrieval algorithms in other work [12, 17, 19, 33]: for each document $d \in \mathcal{D}_{\text{init}}$, we have the cluster $\{d\} \cup Nbhd(d \mid k-1, \mathcal{D}_{\text{init}} - \{d\})$, where k is the cluster-size parameter.

4.2 Experimental Setup

We conducted our experiments on three TREC datasets:

corpus	# of docs	queries	disk(s)
AP	242,918	51-64, 66-150	1-3
TREC8	528,155	401-450	4-5
WSJ	173,252	151-200	1-2

[6] Some of the PageRank results appearing in our previous paper [18] accidentally reflect experiments utilizing a sub-optimal choice of $\mathcal{D}_{\text{init}}$. For citation purposes, the numbers reported in the current paper should be used.

[7] Concatenation order is irrelevant for unigram LMs.

	AP			TREC8			WSJ		
	prec@5	prec@10	MRR	prec@5	prec@10	MRR	prec@5	prec@10	MRR
doc-Auth[d↔d]	.509	.486	.638	.440	.424	.648	.504	.464	.638
doc-PageRank[d↔d]	.519	.480	.632	.524	.446	.666	.536	.486	.699
doc-Auth[c→d]	**.541**	**.501** ^{p}	**.669** ^{p}	**.544** ^{a}	**.452**	**.674**	**.564** ^{a}	**.514** ^{a}	**.746** ^{a}

Table 1: Main comparison: HITS or PageRank on document-only graphs versus HITS on cluster-to-document graphs. Bold: best results per column. Symbols "p" and "a": doc-Auth[c→d] result differs significantly from that of doc-PageRank[d↔d] or doc-Auth[d↔d], respectively.

We applied basic tokenization and Porter stemming via the Lemur toolkit (www.lemurproject.org), which we also used for language-model induction. Topic titles served as queries.

In many retrieval situations of interest, ensuring that the top few documents retrieved (a.k.a., "the first page of results") tend to be relevant is much more important than ensuring that we assign relatively high ranks to the entire set of relevant documents in aggregate [31]. Hence, rather than use mean average precision (MAP) as an evaluation metric, we apply metrics more appropriate to the structural re-ranking task: precision at the top 5 and 10 documents (henceforth prec@5 and prec@10, respectively) and the mean reciprocal rank (MRR) of the first relevant document [31]. All performance numbers are averaged over the set of queries for a given corpus.

The natural baseline for the work described here is the standard language-model-based retrieval approach [29, 5], since it is an effective paradigm that makes no explicit use of inter-document relationships. Specifically, for a given evaluation metric e, the corresponding *optimized baseline* is the ranking on documents produced by $p_d^{[\mu(e)]}(q)$, where $\mu(e)$ is the value of the Dirichlet smoothing parameter that results in the best retrieval performance as measured by e.

A ranking method might assign different items the same score; we break such ties by item ID. Alternatively, the scores used to determine $\mathcal{D}_{\text{init}}$ can be utilized, if available.

Parameter selection for graph-based methods. There are two motivations underlying our approach to choosing values for our algorithms' parameters [18].

First, we hope to show that structural re-ranking can provide better results than the optimized baselines even when initialized with a sub-optimal (yet reasonable) ranking. Hence, let the *initial ranking* be the document ordering induced on the entire corpus by $p_d^{[\mu_{1000}]}(q)$, where μ_{1000} is the smoothing-parameter value optimizing the average non-interpolated precision of the top 1000 documents. We set $\mathcal{D}_{\text{init}}$ to the top 50 documents in the initial ranking.

Second, we wish to show that good results can be achieved without a great deal of parameter tuning. Therefore, we did not tune the smoothing parameter for any of the language models used to determine graph edge-weights, but rather simply set $\mu = 2000$ when smoothing was required, following a prior suggestion [38]. Also, *the other free parameters' values were chosen so as to optimize prec@5, regardless of the evaluation metric under consideration.*[8] As a consequence, our prec@10 and MRR results are presumably not as high as possible; but the advantage of our policy is that we can see whether optimization with respect to a fixed criterion yields good results no matter how "goodness" is measured.

Parameter values were selected from the following sets. The graph "out-degree" δ: $\{2, 4, 9, 19, 29, 39, 49\}$. The cluster size k: $\{2, 5, 10, 20, 30\}$. The PageRank damping factor λ: $\{0.05, 0.1 \ldots 0.9, 0.95\}$.

[8]If two different parameter settings yield the same prec@5, we choose the setting *minimizing* prec@10 so as to provide a conservative estimate of expected performance. Similarly, if we have ties for both prec@5 and prec@10, we choose the setting minimizing MRR.

5. EXPERIMENTAL RESULTS

In what follows, when we say that results or the difference between results are "significant", we mean according to the two-sided Wilcoxon test at a confidence level of 95%.

5.1 Re-Ranking by Document Centrality

Main result. We first consider our main question: can we substantially boost the effectiveness of HITS by applying it to cluster-to-document graphs, which we have argued are more suitable for it than the document-to-document graphs we constructed in our previous work [18]? The answer, as shown in Table 1, is clearly "yes": we see that *moving to cluster-to-document graphs results in substantial improvement for HITS, and indeed boosts its results over those for PageRank on document-to-document graphs.*

Full suite of comparisons. We now turn to Figure 2, which gives the results for the re-ranking algorithms doc-Influx, doc-PageRank and doc-Auth as applied to either the document-based graph d↔d (as in [18]) or the cluster-document graph c→d. (Discussion of doc-Hub is deferred to Section 5.3.)

To focus our discussion, it is useful to first point out that in almost all of our nine evaluation settings (3 corpora × 3 evaluation measures), all three of the re-ranking algorithms perform better when applied to c→d graphs than to d↔d graphs, as the number of dark bars in Figure 2 indicates. Since it is thus clearly useful to incorporate cluster-based information, we will now mainly concentrate on c→d-based algorithms.

The results for prec@5, the metric for which the re-ranking algorithms' parameters were optimized, show that *all* c→d-based algorithms outperform the prec@5-optimized baseline — significantly so for the AP corpus — even though applied to a sub-optimally-ranked initial set. (We hasten to point out that while the initial ranking is always inferior to the corresponding optimized baseline, the differences are never significant.) In contrast, the use of d↔d graphs never leads to significantly superior prec@5 results.

We also observe in Figure 2 that the doc-Auth[c→d] algorithm is always either the best of the c→d-based algorithms or clearly competitive with the best. Furthermore, pairwise comparison of it to each of the doc-Influx[c→d]

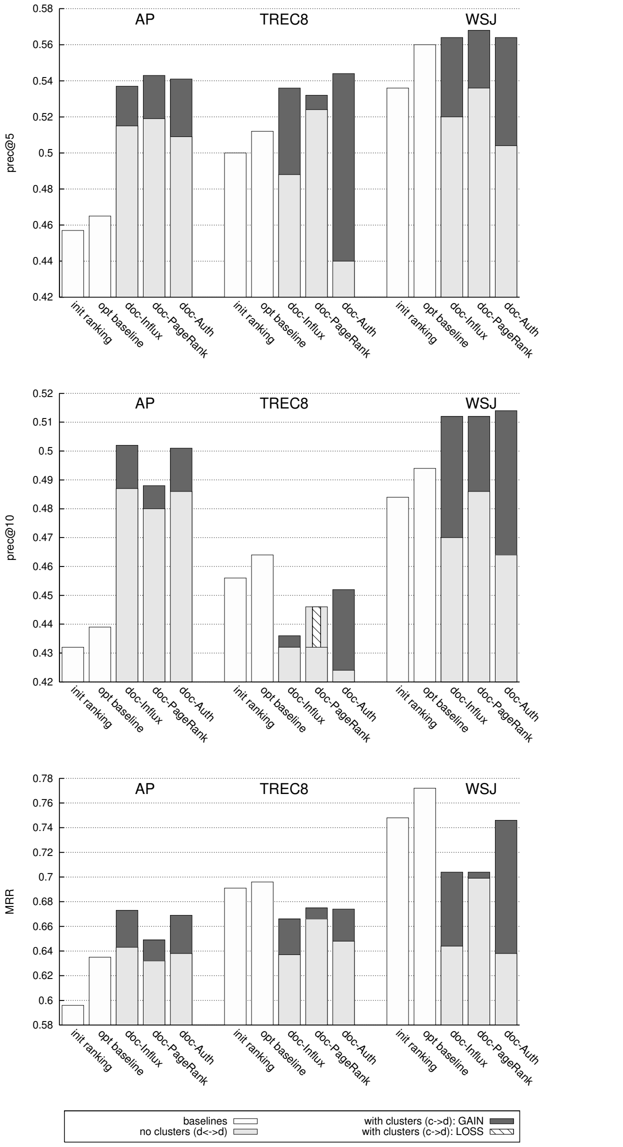

Figure 2: All re-ranking algorithms, as applied to either d↔d graphs or c→d graphs.

and doc-PageRank[c→d] algorithms favors the HITS-style doc-Auth[c→d] algorithm in a majority of the evaluation settings.

We also experimented with a few alternate graph-construction methods, such as sum-normalizing the weights of edges out of nodes, and found that the doc-Auth[c→d] algorithm remained superior to doc-Influx[c→d] and doc-PageRank[c→d]. We omit these results due to space constraints.

All in all, these findings lead us to believe that not only is it useful to incorporate information from clusters, but it can be more effective to do so in a way reflecting the mutually-reinforcing nature of clusters and documents, as the HITS algorithm does.

5.2 Re-Ranking by Cluster Centrality

We now consider the alternative, mentioned in Section 2.4, of using the centrality scores for *clusters* as an *indirect* means of ranking documents, in the sense of identifying clusters that contain a high percentage of relevant documents. Note that the problem of automatically identifying such clusters in the re-ranking setting has been acknowledged to be a hard task for some time [36]. Nevertheless, as stated in Section 2.4, we experimented with Liu and Croft's general clusters-for-selection approach [25]: rank the clusters, then rank the documents within each cluster by $p_d^{[\mu]}(q)$. Our baseline algorithm, clust-$p_c^{[\mu]}(q)$, adopts Liu and Croft's specific proposal of the CQL algorithm — except that we employ overlapping rather than hard clusters — wherein clusters are ranked by the *query likelihood* $p_c^{[\mu]}(q)$ instead of one of our centrality scores.

Table 2 (which may appear on the next page) presents the performance results. Our first observation is that the clust-Influx[d→c] and clust-Auth[d→c] algorithms are superior in a majority of the relevant comparisons to the initial ranking, the optimized baselines, and the clust-$p_c^{[\mu]}(q)$ algorithm, where the performance differences with the latter sometimes achieve significance.

However, the performance of the document-centrality-based algorithm doc-Auth[c→d] is better in a majority of the evaluation settings than that of any of the cluster-centrality-based algorithms. On the other hand, it is possible that the latter methods could be improved by a better technique for within-cluster ranking.

To compare the effectiveness of clust-Influx[d→c] and clust-Auth[d→c] to that of clust-$p_c^{[\mu]}(q)$ in detecting clusters with a high percentage of relevant documents — thereby neutralizing within-cluster ranking effects — we present in Table 3 the percent of documents in the highest ranked cluster that are relevant. (Cluster size (k) was fixed to either 5 or 10 and out-degree (δ) was chosen to optimize the above percentage.) Indeed, these results clearly show that our best cluster-based algorithms are much better than clust-$p_c^{[\mu]}(q)$ in detecting clusters containing a high percentage of relevant documents, in most cases to a significant degree.

Cluster ranking	AP		TREC8		WSJ	
	k=5	k=10	k=5	k=10	k=5	k=10
$p_c^{[\mu]}(q)$	39.2	38.8	39.6	40.6	44.0	37.0
Influx[d→c]	48.7 ^{c}	**47.6** ^{c}	48.0	43.8	51.2 ^{c}	48.0 ^{c}
Auth[d→c]	**49.5** ^{c}	47.2 ^{c}	**50.8** ^{c}	**46.6**	**53.6** ^{c}	**49.0** ^{c}

Table 3: Average relevant-document percentage within the top-ranked cluster. k: cluster size. Bold: best results per column. c: result differs significantly from that of clust-$p_c^{[\mu]}(q)$, used in [25].

5.3 Further Analysis

Authorities versus hubs. So far, we have only considered utilizing the authority scores that the HITS algorithm produces. The chart below shows the effect of ranking entities by hub scores instead. Specifically, the "documents?" column compares doc-Auth[c→d] (i.e., ranking documents by authoritativeness) to doc-Hub[d→c] (i.e., ranking documents by hubness); similarly, the "clusters?" column compares clust-Auth[d→c] to clust-Hub[c→d]. Each entry depicts, in descending order of performance (except for the one indicated tie) as one moves left to right, those central-

	AP			TREC8			WSJ		
	prec@5	prec@10	MRR	prec@5	prec@10	MRR	prec@5	prec@10	MRR
init. ranking	.457	.432	.596	.500	.456	.691	.536	.484	.748
opt. baselines	.465	.439	.635	.512	.464	.696	.560	.494	**.772**
clust-$p_c^{[\mu]}(q)$	.448	.418	.549 ^{io}	.500	.432	**.723**	.504 ^{o}	.454 ^{io}	.680
clust-Influx[d→c]	.511 ^{c}	**.479** ^{c}	.619	.524	**.478**	.681	**.568** ^{c}	**.512** ^{c}	.760
clust-PageRank[d→c]	.493	.475 ^{c}	.595	.496	.444	.683	.528	.490 ^{c}	.736
clust-Auth[d→c]	**.533** ^{ioc}	.478 ^{c}	**.651** ^{c}	**.532**	.460	.714	.552	.478	.757

Table 2: Cluster-based re-ranking. Bold: best results per column. Symbols i, o, c: results differ significantly from the initial ranking, optimized baseline, or (for the re-ranking algorithms) clust-$p_c^{[\mu]}(q)$ [25], respectively.

ity scoring functions that lead to an improvement over the initial ranking: A stands for "authority" and H for "hub". Cases in which the improvement is significant are marked with a '*'.

When do we improve the initial ranking by measuring the centrality of:			
		documents?	clusters?
AP	prec @5	A^*H	A^*H
	prec @10	A^*H	AH
	MRR	AH	A
TREC8	prec @5	AH	AH
	prec @10		HA
	MRR	H	H^*A
WSJ	prec @5	AH	AH (tie)
	prec @10	AH	H
	MRR		HA

We see that in many cases, hub-based re-ranking does yield better performance than the initial ranking. But authority-based re-ranking appears to be an even better choice overall.

HITS on PageRank-style graphs. Consider our comparison of doc-Auth[d↔d] against doc-PageRank[d↔d]. As the notation suggests, this corresponds to running HITS and PageRank on the same graph, d↔d. But an alternative interpretation [18] is that non-smoothed (or no-random-jump) PageRank, as expressed by Equation (3), is applied to a *different* version of d↔d wherein the original edge weights $wt(u \rightarrow v)$ have been smoothed as follows:

$$wt^{[\lambda]}(u \rightarrow v) \stackrel{def}{=} \frac{1-\lambda}{|V|} + \lambda \frac{wt(u \rightarrow v)}{out(u)} \quad (7)$$

(we ignore nodes with no positive-weight out-edges to simplify discussion, and omit the d↔d superscripts for clarity).

How does HITS perform on document-to-document graphs that are "truly equivalent", in the sense of employing the above edge-weighting regime, to those that PageRank is applied to? One reason this is an interesting question is that HITS assigns scores of zero to nodes that are not in the graph's largest connected component (with respect to positive-weight edges, considered to be bi-directional). Notice that the original graph may have several connected components, whereas utilizing $wt^{[\lambda]}$ ensures that each node has a positive-weight directed edge to every other node. Additionally, the re-weighted version of HITS has provable stability properties [27].

We found that in nearly all of our evaluation settings for document-to-document graphs (three corpora × three evaluation metrics), doc-Auth[d↔d] achieved better results using $wt^{[\lambda]}$ edge weights. However, we cannot discount the possibility that the performance differences might be due simply to the inclusion of the extra interpolation-parameter λ. Moreover, in all but one case, the improved results were still below those for doc-PageRank[d↔d] (and always lagged behind those of doc-Auth[c→d]).

Interestingly, the situation is qualitatively different if we consider c→d graphs instead. In brief, we applied a smoothing scheme analogous to that described above, but only to edges leading from a left-hand node (cluster) to a right-hand node (document)[9]; we thus preserved the one-way bipartite structure. Only in two of the nine evaluation settings did this change cause an increase in performance of doc-Auth[c→d] over the results attained under the original edge-weighting scheme, despite the fact that the re-weighting involves an extra free parameter. Thus, while we have already demonstrated in previous sections of this paper that information about document-cluster similarity relationships is very valuable, the results just mentioned suggest that such information is more useful in "raw" form.

Re-anchoring to the query. In previous work, we showed that PageRank centrality scores induced over document-based graphs can be used as a multiplicative weight on document query-likelihood terms, the intent being to cope with cases in which centrality in $\mathcal{D}_{\text{init}}$ and relevance are not strongly correlated [18]. Indeed, employing this technique on the AP, TREC8, and WSJ corpora, prec@5 increases from .519, .524 and .536, to .531, .56 and .572 respectively.

The same modification could be applied to the c→d-based algorithms, although it is not particularly well-motivated in the HITS case. While PageRank scores correspond to a stationary distribution that could be loosely interpreted as a prior [18], in which case multiplicative combination with query likelihood is sensible, it is not usual to assign a probabilistic interpretation to hub or authority scores.

Nonetheless, for the sake of comparison completeness, we applied this idea to the doc-Auth[c→d] algorithm, yielding the following performance changes: from .541, .544, and .564 to .537, .572 and .572 respectively. These results are still as good as — and for two corpora better than — those for PageRank as a multiplicative weight on query likelihood. Thus, it may be the case that centrality scores induced over a document-based graph are more effective as a multiplicative bias on query-likelihood than as direct representations of rel-

[9]In the one-way bipartite case, the "$|V|$" in Equation (7) must be changed to the number of right-hand nodes.

evance in $\mathcal{D}_{\text{init}}$ (see also [18]); but, modulo the caveat above, it seems that when centrality is induced over cluster-based one-way bipartite graphs, the correlation with relevance is much stronger, and hence this kind of centrality serves as a better "bias" on query-likelihood.

6. CONCLUSION

We have shown that leveraging the mutually reinforcing relationship between clusters and documents to determine centrality is very beneficial not only for directly finding relevant documents in an initially retrieved list, but also for finding clusters of documents from this list that contain a high number of relevant documents.

Specifically, we demonstrated the superiority of cluster-document bipartite graphs to document-only graphs as the input to centrality-induction algorithms. Our method for finding "authoritative" documents (or clusters) using HITS over these bipartite graphs results in state-of-the-art performance for document (and cluster) re-ranking.

Acknowledgments We thank Eric Breck, Claire Cardie, Oren Etzioni, Jon Kleinberg, Art Munson, Filip Radlinski, Ves Stoyanov, Justin Wick and the anonymous reviewers for valuable comments. This paper is based upon work supported in part by the National Science Foundation under grant no. IIS-0329064 and an Alfred P. Sloan Research Fellowship. Any opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views or official policies, either expressed or implied, of any sponsoring institutions, the U.S. government, or any other entity.

7. REFERENCES

[1] J. Baliński and C. Daniłowicz. Re-ranking method based on inter-document distances. *Information Processing and Management*, 41(4):759–775, 2005.

[2] D. Beeferman and A. L. Berger. Agglomerative clustering of a search engine query log. In *Proceedings of KDD*, pages 407–416, 2000.

[3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th International World Wide Web Conference*, pages 107–117, 1998.

[4] W. B. Croft. A model of cluster searching based on classification. *Information Systems*, 5:189–195, 1980.

[5] W. B. Croft and J. Lafferty, editors. *Language Modeling for Information Retrieval*. Number 13 in Information Retrieval Book Series. Kluwer, 2003.

[6] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/Gather: A cluster-based approach to browsing large document collections. In *15th Annual International SIGIR*, pages 318–329, Denmark, June 1992.

[7] C. Daniłowicz and J. Baliński. Document ranking based upon Markov chains. *Information Processing and Management*, 41(4):759–775, 2000.

[8] I. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the Seventh ACM SIGKDD Conference*, pages 269–274, 2001.

[9] F. Diaz. Regularizing ad hoc retrieval scores. In *Proceedings of the Fourteenth International Conference on Information and Knowledge Managment (CIKM)*, pages 672–679, 2005.

[10] G. Erkan. Language model based document clustering using random walks. In *Proceedings of HLT/NAACL*, 2006.

[11] G. Erkan and D. R. Radev. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004.

[12] A. Griffiths, H. C. Luckhurst, and P. Willett. Using interdocument similarity information in document retrieval systems. *Journal of the American Society for Information Science (JASIS)*, 37(1):3–11, 1986. Reprinted in Karen Sparck Jones and Peter Willett, eds., *Readings in Information Retrieval*, Morgan Kaufmann, pp. 365–373, 1997.

[13] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of SIGIR*, 1996.

[14] N. Jardine and C. J. van Rijsbergen. The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval*, 7(5):217–240, 1971.

[15] Y. Karov and S. Edelman. Similarity-based word sense disambiguation. *Computational Linguistics*, 24(1):41–59, 1998.

[16] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 668–677, 1998. Extended version in *Journal of the ACM, 46:604–632, 1999*.

[17] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of SIGIR*, pages 194–201, 2004.

[18] O. Kurland and L. Lee. PageRank without hyperlinks: Structural re-ranking using links induced by language models. In *Proceedings of SIGIR*, pages 306–313, 2005.

[19] O. Kurland, L. Lee, and C. Domshlak. Better than the real thing? Iterative pseudo-query processing using cluster-based language models. In *Proceedings of SIGIR*, pages 19–26, 2005.

[20] J. D. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of SIGIR*, pages 111–119, 2001.

[21] A. N. Langville and C. D. Meyer. Deeper inside PageRank. *Internet Mathematics*, 2005.

[22] A. Leuski. Evaluating document clustering for interactive information retrieval. In *Proceedings of the Tenth International Conference on Information and Knowledge Managment (CIKM)*, pages 33–40, 2001.

[23] A. Leuski and J. Allan. Evaluating a visual navigation system for a digital library. In *Proceedings of the Second European conference on research and advanced technology for digital libraries (ECDL)*, pages 535–554, 1998.

[24] G.-A. Levow and I. Matveeva. University of Chicago at CLEF2004: Cross-language text and spoken document retrieval. In *Proceedings of CLEF*, pages 170–179, 2004.

[25] X. Liu and W. B. Croft. Cluster-based retrieval using language models. In *Proceedings of SIGIR*, pages 186–193, 2004.

[26] R. Mihalcea and P. Tarau. TextRank: Bringing order into texts. In *Proceedings of EMNLP*, pages 404–411, 2004. Poster.

[27] A. Y. Ng, A. X. Zheng, and M. I. Jordan. Stable algorithms for link analysis. In *Proceedings of SIGIR*, pages 258–266, 2001.

[28] J. Otterbacher, G. Erkan, and D. R. Radev. Using random walks for question-focused sentence retrieval. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pages 915–922, 2005.

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

[30] S. E. Preece. Clustering as an output option. In *Proceedings of the American Society for Information Science*, pages 189–190, 1973.

[31] C. Shah and W. B. Croft. Evaluating high accuracy retrieval techniques. In *Proceedings of SIGIR*, pages 2–9, 2004.

[32] X. Shen and C. Zhai. Active feedback in ad hoc information retrieval. In *Proceedings of SIGIR*, pages 59–66, 2005.

[33] T. Tao, X. Wang, Q. Mei, and C. Zhai. Language model information retrieval with document expansion. In *Proceedings of HLT/NAACL*, 2006.

[34] A. Tombros, R. Villa, and C. van Rijsbergen. The effectiveness of query-specific hierarchic clustering in information retrieval. *Information Processing and Management*, 38(4):559–582, 2002.

[35] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, second edition, 1979.

[36] P. Willett. Query specific automatic document classification. *International Forum on Information and Documentation*, 10(2):28–32, 1985.

[37] O. Zamir and O. Etzioni. Web document clustering: a feasibility demonstration. In *Proceedings of SIGIR*, pages 46–54, 1998.

[38] C. Zhai and J. D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR*, pages 334–342, 2001.

[39] B. Zhang, H. Li, Y. Liu, L. Ji, W. Xi, W. Fan, Z. Chen, and W.-Y. Ma. Improving web search results using affinity graph. In *Proceedings of SIGIR*, pages 504–511, 2005.