On Identifying Representative Relevant Documents

Fiana Raiber
fiana@tx.technion.ac.il

Oren Kurland
kurland@ie.technion.ac.il

Faculty of Industrial Engineering and Management
Technion — Israel Institute of Technology
Haifa 32000, Israel

ABSTRACT

Using relevance feedback can significantly improve the effectiveness of ad hoc (query-based) retrieval. However, retrieval performance can significantly vary with respect to the given set of relevant documents. Our goal is to establish a quantitative analysis of what makes a relevant document a good *representative* of the relevant-documents set *regardless* of the retrieval approach employed. That is, we would like to estimate the extent to which a relevant document can effectively help in finding (other) relevant documents using *some* relevance-feedback method employed over the corpus. We present various representativeness estimates; some of which treat documents independently and some utilize inter-document similarities. Empirical evaluation shows that relevant documents that are centrally located within the similarity space of the relevant-documents set tend to be good representatives. In addition, we show that there exist highly representative*clusters* of similar relevant documents, and devise methods for ranking clusters based on their presumed representativeness. Finally, we study the connection between representativeness and TREC's gradual relevance judgments.

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

General Terms: Algorithms, Experimentation

Keywords: ad hoc retrieval, learning to rank, relevance feedback, representative clusters, representative relevant documents

1. INTRODUCTION

The ad hoc retrieval task is finding documents in a corpus that pertain to information need expressed by a query. The task becomes easier if *relevance feedback* is available (e.g., examples of relevant documents). Indeed, there is a large body of work on devising retrieval methods that exploit information induced from relevant (and non-relevant) documents [27, 11, 20, 39, 28, 3, 25, 36].

However, some previous work showed that the effectiveness of relevance-feedback-based retrieval can significantly vary with respect to the given set of relevant documents [9, 37, 33]. Specifically, there are many cases wherein using some relevant documents degrades retrieval performance *regardless* of the retrieval method used [37, 33]. Yet, the question of which relevant documents are more effective than others for relevance-feedback-based retrieval has remained an open challenge.

Thus, we aim to establish a quantifiable characterization of the degree to which relevant documents can be effectively used in relevance-feedback-based retrieval *regardless* of the retrieval approach that is employed. To that end, we set the following task as a goal. Given a query, and a set of documents relevant to the information need it expresses, we want to automatically identify the best "*representatives*" of the set. We consider a relevant document as a good representative to the extent it can effectively help to find documents from the relevant-documents set via a search performed over the corpus. More specifically, we want to identify a subset of k relevant documents that if fed with the query to *some* relevance feedback algorithm [27, 28, 3] used to rank the entire corpus, then the resultant performance is optimal with respect to subsets of k relevant documents.

We present two paradigms of identifying the most representative relevant documents. The first is ranking relevant documents using estimates of their representativeness, and selecting the top-ranked ones. Some of the estimates we propose consider documents independently while others utilize inter-document similarities. We also explore a learning-to-rank [22] approach that integrates the estimates.

The second paradigm is based on the premise, which could be viewed as an extension of the *cluster hypothesis* [35], that representative relevant documents are similar to each other. Accordingly, we devise methods for ranking *clusters* of relevant documents based on their presumed representativeness — i.e., the resultant performance of a relevance feedback method when fed with the cluster's constituent documents. The documents in the highest ranked cluster are then considered as the best representatives.

Experiments performed over TREC corpora using two state-of-the-art relevance-feedback methods show that relevant documents that are centrally located within the similarity space of the relevant-documents set are often good representatives. Furthermore, we show that there are highly representative clusters of similar relevant documents, but automatically identifying these is a hard challenge. A learning-to-rank method for clusters is the most effective among those we consider for identifying representative clusters.

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.

Further exploration shows that documents considered as "bad" representatives by our methods tend to be "poison pills" [33]; that is, if used alone (with the query) in relevance-feedback-based retrieval, then the resultant performance is worse than that of not utilizing relevance feedback at all.

In addition, we show that levels of representativeness induced by our methods are connected with TREC's levels of relevance (relevant vs. highly relevant). Hence, the methods could potentially be viewed as a means for creating (one type) of gradual relevance judgments. However, it turns out that while document-query surface level similarity is among the worst representativeness estimates that we study, it is a highly effective indicator of relevant documents that may be regarded as "highly relevant" by TREC's judges.

Finally, it is important to differentiate our goal here from that of devising (i) a relevance feedback method, and (ii) a method for selecting which documents should be presented to the user to provide feedback on [30]. Furthermore, some of the estimates that we present utilize the entire set of relevant documents, which is obviously not available during retrieval time. As stated above, our focus is on providing a better understanding — specifically, a quantitative analysis — of what makes relevant documents (in)effective for relevance-feedback-based retrieval.

2. IDENTIFYING REPRESENTATIVE RELEVANT DOCUMENTS

Throughout this section we assume that the following have been fixed. A query q, a corpus of documents $\mathcal{D}$, and the set $\mathcal{D}_{rel}^{[q;n]}$ (henceforth $\mathcal{D}_{rel}$; $\mathcal{D}_{rel} \subset \mathcal{D}$) of *all* n documents in the corpus that are relevant to the information need expressed by q.[1]

Our goal is to identify the k ($< n$) documents in $\mathcal{D}_{rel}$ that are its best "*representatives*". We consider a relevant document as a good representative of $\mathcal{D}_{rel}$ to the extent it can "effectively help" to find relevant documents from $\mathcal{D}_{rel}$ via a search performed over the corpus. More specifically, the task is identifying the k documents in $\mathcal{D}_{rel}$ that if used as input, along with the query, to *some* relevance feedback algorithm that is used to rank the *entire* corpus, then the resultant retrieval performance will be optimal with respect to all subsets of k documents in $\mathcal{D}_{rel}$.

In what follows, we propose two paradigms for addressing our goal. The first, presented in Section 2.1, is ranking $\mathcal{D}_{rel}$ by the presumed representativeness of its documents, and selecting the top-k ranked documents. The second paradigm, presented in Section 2.2, is based on the premise that representatives are similar to each other. Specifically, we cluster $\mathcal{D}_{rel}$ into clusters of k similar documents, rank the clusters based on their *presumed* representativeness — i.e., the resultant performance of a relevance feedback algorithm when fed with the cluster's constituent documents — and then select the documents in the highest ranked cluster.

The methods we present utilize language models [26, 6]. We use $p(w|x)$ to denote the probability assigned by a language model induced from the text (or text collection) x to term w; by definition, $p(\cdot|x)$ is a probability distribution over the vocabulary. Unless otherwise stated, we assume that the language model is smoothed. Some of our methods also utilize $sim(x, y)$ — a language-model-based measure of the similarity between texts x and y. Details regarding our language-model induction technique and similarity measure are provided in Section 4.1.1.

[1]In the evaluation presented in Section 4 this set contains all documents marked as relevant by TREC's annotators.

2.1 Ranking relevant documents based on presumed representativeness

The following methods assign document d ($\in \mathcal{D}_{rel}$) with a score, $Score(d)$, that presumably reflects the extent to which d is a good representative of $\mathcal{D}_{rel}$. The top-k scoring documents in $\mathcal{D}_{rel}$ are then used, along with the query, as input to a relevance-feedback method[2].

2.1.1 Considering documents independently

The first family of methods that we consider estimates d's representativeness independently of other documents in $\mathcal{D}_{rel}$.

As a reference comparison to all proposed approaches, we use the **Random** method that scores d by

$$Score_{Random}(d) \stackrel{def}{=} \frac{1}{n};$$

then, k documents from $\mathcal{D}_{rel}$ are randomly selected.

As $\mathcal{D}_{rel}$ is a set of relevant documents, we consider the **QuerySim** method that regards document d that exhibits high (surface-level) query similarity as a good representative:

$$Score_{QuerySim}(d) \stackrel{def}{=} sim(q, d).$$

The **Length** method is based on the assumption that short relevant documents are better representatives than long relevant documents:

$$Score_{Length}(d) \stackrel{def}{=} -|d|;$$

$|d|$ is the number of terms in d. A case in point: the ratio between the volume of relevant and non-relevant information in a short document is potentially larger than that in a longer document.

Along the same lines, we hypothesize that documents with homogeneous content (i.e., that focus on a single/few topics/themes) are potentially more representative than those containing heterogeneous content. To quantify this notion, we use the **entropy** of d's unsmoothed language model,

$$Score_{Entropy}(d) \stackrel{def}{=} -H(p(\cdot|d)),$$

where $H(p(\cdot|d)) \stackrel{def}{=} -\sum_w p(w|d) \log p(w|d)$. Low values of entropy imply that the term-frequency distribution is concentrated around a relatively small number of terms, which in turn potentially implies to content homogeneity [18].

Another potential indicator of the content-homogeneity of a document is its **clarity** with respect to the corpus. That is, we hypothesize that the more different a document (unsmoothed) language model is from that of the corpus — which could be viewed as a general non-relevant document — the more "focused/clear" the document; consequently, the better representative of $\mathcal{D}_{rel}$ the document is assumed to be. We use the KL-divergence (D) to measure this language-model difference: $D\left(p(\cdot|d) \,\middle|\middle|\, p(\cdot|\mathcal{D})\right) \stackrel{def}{=}$

[2]We assume that the documents' scores are not provided to the relevance-feedback method.

$\sum_w p(w|d) \log \frac{p(w|d)}{p(w|\mathcal{D})}$, so as to derive a clarity-based representativeness criterion:

$$Score_{Clarity}(d) \stackrel{def}{=} D\left(p(\cdot|d) \,\middle|\middle|\, p(\cdot|\mathcal{D})\right).$$

(High KL-divergence means large difference between the language models.) Originally, the notion of clarity was presented for queries so as to predict query performance [7].

2.1.2 Exploiting inter-document relationships

We now turn to study methods that estimate representativeness by utilizing (similarity) relationships between documents in $\mathcal{D}_{rel}$. In contrast to the approaches from above that use only information in the document (and some corpus statistics), these methods require a knowledge of the entire set of relevant documents.

The centroid method. A natural approach to estimating document d's representativeness is measuring its similarity to some representation of the entire set $\mathcal{D}_{rel}$ — e.g., $\mathcal{D}_{rel}$'s centroid, $Cent(\mathcal{D}_{rel})$. In language model terms, the centroid is a probability distribution over the vocabulary: $p(w|Cent(\mathcal{D}_{rel})) \stackrel{def}{=} \frac{1}{n}\sum_{d' \in \mathcal{D}_{rel}} p(w|d')$. Then, the **Centroid** method estimates d's representativeness using the KL-divergence of its induced language model from the centroid:

$$Score_{Centroid}(d) \stackrel{def}{=} -D\left(p(\cdot|Cent(\mathcal{D}_{rel})) \,\middle|\middle|\, p(\cdot|d)\right);$$

the lower the KL-divergence, the closer the document language model is to the centroid.

Graph-based methods. Some work on re-ranking search results [8, 18] has demonstrated the merits of utilizing information induced from inter-document similarities. Specifically, given an initial list of documents retrieved in response to a query, documents that are highly similar to many other documents in the list were shown to have high probability of relevance [18]. The idea is that these documents represent the entire list, and by the virtue of the way the list was created — i.e., in response to the query — they might be relevant to the underlying information need. However, the list is composed of both relevant and non-relevant documents.

Here, we opt to adapt the idea just described to the case of the set $\mathcal{D}_{rel}$, which contains only relevant documents. Specifically, we hypothesize that a relevant document in $\mathcal{D}_{rel}$ is representative to the extent it is similar to other (representative) documents in $\mathcal{D}_{rel}$. To quantify these representativeness notions, we employ previously-proposed graph-based approaches [18].

Let $G = (\mathcal{D}_{rel}, \mathcal{D}_{rel} \times \mathcal{D}_{rel})$ be the complete directed[3] graph defined over $\mathcal{D}_{rel}$. The weight $wt(d_1 \rightarrow d_2)$ of the edge between d_1 and d_2 is defined as

$$wt(d_1 \rightarrow d_2) \stackrel{def}{=} \begin{cases} sim(d_1, d_2) & \text{if } d_2 \in Nbhd(d_1; \delta), \\ 0 & \text{otherwise,} \end{cases}$$

where $Nbhd(d_1; \delta)$ is the set of δ documents $d' \in \mathcal{D}_{rel} - \{d_1\}$ that yield the highest $sim(d_1, d')$ — i.e., d_1's nearest-neighbors in the similarity space.

The weighted in-degree criterion for representativeness, **WInDeg**, is based on the premise that a document is representative to the extent it gets similarity-based "support" from other documents:

$$Score_{WInDeg}(d) \stackrel{def}{=} \sum_{d' \in \mathcal{D}_{rel}} wt(d' \rightarrow d).$$

To further reward documents that get similarity-based support from documents that are, themselves, representatives to a good extent, which leads to a recursive definition of representativeness, we use PageRank's (**WPR**) approach [2]. (The "W" stands for using weighted edges.) That is, we smooth the edge-weight function:

$$wt^{[\nu]}(d_1 \rightarrow d_2) \stackrel{def}{=} \nu \cdot \frac{1}{|\mathcal{D}_{rel}|} + (1-\nu) \cdot \frac{wt(d_1 \rightarrow d_2)}{\sum_{d' \in \mathcal{D}_{rel}} wt(d_1 \rightarrow d')};$$

ν is a free parameter. Thus, G with the edge-weight function $wt^{[\nu]}$ constitutes an ergodic Markov chain, for which a stationary distribution, $\mathcal{P}(\cdot)$, exists. We set

$$Score_{WPR}(d) \stackrel{def}{=} \mathcal{P}(d).$$

For completeness, we also consider variants of the WInDeg and WPR representativeness measures, denoted **UInDeg** and **UPR**, respectively, that use uniform edge weights [18]:

$$wt_{uniform}(d_1 \rightarrow d_2) \stackrel{def}{=} \begin{cases} 1 & \text{if } d_2 \in Nbhd(d_1; \delta), \\ 0 & \text{otherwise.} \end{cases}$$

Thus, UInDeg estimates d's representativeness by the number of documents in $\mathcal{D}_{rel}$ that d is among the δ nearest-neighbors of. UPR is the analog of the original PageRank formula, wherein edges with a uniform, non-zero, weight are drawn between documents and their δ nearest neighbors.

Diversity. Recall that our goal is to select a set of k representative relevant documents that will help in finding relevant documents when using relevance-feedback-based search. An hypothesis that we turn to explore now is that the selected documents should be different (content wise) from each other, so as to potentially reflect diversified aspects of relevance. To that end, we use the Maximal Marginal Relevance criterion [5], **MMR**. Specifically, a document is scored by its similarity to the query, and dissimilarity with documents already selected (S):

$Score_{MMR}(d) \stackrel{def}{=} \psi sim(q, d) - (1-\psi) \max_{d_i \in S} sim(d, d_i)$; ψ is a free parameter; the first selected document is that exhibiting the highest query similarity, $sim(q, d)$.

2.1.3 Learning to rank relevant documents

To integrate the different estimates for representativeness presented insofar, we take a *learning-to-rank*, **LTR**, approach [22]. Specifically, we use the standard SVM^{rank} method [15] that given examples of queries and rankings of documents for these queries, learns a model that can be used to rank documents for new queries. Here, for each query we provide the learner with a ranking of its relevant documents. The ranking is determined based on the retrieval effectiveness (specifically, as measured by mean average precision (MAP)) of the relevance-feedback method when fed with the query and a (single) document from $\mathcal{D}_{rel}$.

Documents in SVM^{rank} are represented by vectors of features. In our case, these features should attest to the potential representativeness of the document. Specifically, we use

[3]There is some work on the importance of directionality of edges in such graphs [16].

the QuerySim, Length, Entropy, Clarity, Centroid, WInDeg, WPR, UInDeg, and UPR values for a document.[4]

2.2 Finding representative clusters of similar relevant documents

The methods presented above assign a representativeness score to each document; then, the top-k scoring documents are selected. While, in general, no constraint has been posted on the relationships between the selected documents, the graph-based methods are based on the implicit premise that these documents should be similar to each other. We further explore this idea by explicitly posting an inter-document-similarity constraint.

The principle of searching for k similar representative documents could be regarded as an operational extension of the *cluster hypothesis* [35]. The hypothesis states that "closely-associated documents tend to be relevant to the same requests", and is often translated to statements about the similarity between relevant documents being stronger than that between non-relevant documents, and than that between relevant and non-relevant documents [13]. Here, we postulate that inter-document-similarities between representative relevant documents are stronger than those between non-representative relevant documents, and those between representative and non-representative relevant documents.

To devise a concrete method based on the hypothesis just stated, we cluster $\mathcal{D}_{rel}$ into clusters of k documents. Specifically, we use a simple nearest-neighbor clustering approach wherein we define for each d ($\in \mathcal{D}_{rel}$) a cluster c_d that contains d and the $k-1$ documents d' ($d' \neq d$) in $\mathcal{D}_{rel}$ that yield the highest $sim(d, d')$. Such overlapping nearest-neighbor clusters are often used in work on cluster-based retrieval [10, 19, 23, 24].

Given the clusters, we opt to find the *one* that is the most representative — i.e., that using its constituent documents, along with the query, as input to a relevance feedback algorithm will yield the best performance with respect to all clusters. Our goal is then to rank the clusters based on their presumed representativeness.

We first consider a simple approach, **clust-$\mathcal{M}$**, that assigns cluster c with the mean representativeness score of its constituent documents:

$$Score_{clust-\mathcal{M}}(c) \stackrel{def}{=} \frac{1}{k} \sum_{d \in c} Score_{\mathcal{M}}(d); \quad (1)$$

$\mathcal{M}$ is one of the representativeness models from Sections 2.1.1 and 2.1.2, except for Random and MMR[5]; **clust-Random** will denote random selection of a cluster. Hence, the k documents selected for relevance feedback — i.e., those that belong to the highest ranked cluster — exhibit two properties. Their average representativeness score is high, and they are similar to each other by the virtue of belonging to the same cluster.

Another cluster ranking approach that we examine is based on the hypothesis that the more the cluster is *dense* the better representative it is. We study two measures of density: the average similarity between the constituent documents and the basis-document used to create the cluster (**clust-BaseDense**):

$$Score_{clust-BaseDense}(c_d) \stackrel{def}{=} \frac{1}{k-1} \sum_{d_i \in c_d; d_i \neq d} sim(d, d_i),$$

and the average similarity between all documents in the cluster (**clust-AvgDense**):[6]

$$Score_{clust-AvgDense}(c_d) \stackrel{def}{=} \frac{2}{k(k-1)} \sum_{d_i, d_j \in c_d; i<j} sim(d_i, d_j).$$

2.2.1 Learning to rank clusters of relevant documents

To integrate the cluster ranking methods from above, we employ a learning-to-rank approach, **clust-LTR**, using SVM^{rank} [15]. In the training phase, we order the clusters for each query with respect to the MAP performance obtained by the relevance-feedback algorithm when fed with the query and the cluster's constituent documents. To represent cluster c, we use for features the values $Score_{clust-\mathcal{M}}(c)$, where $\mathcal{M} \in$ {QuerySim, Length, Entropy, Clarity, Centroid, WInDeg, WPR, UInDeg, UPR}, and the two cluster-density values assigned by the measures from above. Normalization of feature values is performed as for documents in Section 2.1.3.

3. RELATED WORK

There is much work on devising effective relevance feedback methods (e.g., [27, 11, 20, 39, 28, 4, 3, 25, 36]). In contrast, we use *some* relevance feedback method(s) to evaluate the representativeness of relevant documents.

Some previous work showed that there exist relevant documents that are ineffective for use in relevance-feedback-based retrieval [37, 33]. However, a quantifiable characterization of such documents has remained an open question. We show in Section 4.2.3 that documents considered as "bad" representatives by our methods tend to be ineffective for relevance-feedback-based retrieval.

One of our goals is to rank *clusters of relevant documents* by their presumed representativeness. This is reminiscent of the *optimal cluster* detection task: finding clusters of top-retrieved documents that contain a high percentage of relevant documents [12, 34, 19, 23, 24, 17]. It was shown that graph-based centrality and query similarity of the cluster's constituent documents are among the most effective indicators of the percentage of relevant documents in the cluster [17, 24]. As noted in Section 2, we use these information sources, and others, to detect representative documents and clusters. However, we show in Section 4.2 that query similarity is a relatively weak indicator for representativeness.

It is important to differentiate the representatives-selection task that we pursue here from those of *active relevance feedback* [30] and *aspect coverage* [38]. The active relevance feedback task [30] is selecting a subset of documents from those most highly ranked by an initial search (both relevant and non relevant) to have the user provide feedback on. The aspect coverage problem [38] is finding a subset of relevant documents that covers many query-related aspects that were defined by human annotators. An effective strategy for both tasks is to select a document set that exhibits high content

[4]For each query we use a min-max normalization for the values assigned by an estimate to documents in $\mathcal{D}_{rel}$. Experiments — results of which are omitted as they convey no additional insight — show that this normalization yields performance superior to that of using the raw values.

[5]Note that the MMR criterion stands in contrast to the goal of finding clusters of similar representative documents.

[6]We assume that documents are assigned with unique sortable IDs.

diversity. However, this strategy, as manifested in our MMR method, is often inferior to other methods that we consider.

4. EVALUATION

In what follows we present an evaluation of the different methods for identifying representative relevant documents. We then study the connection between representativeness and TREC's gradual relevance judgments.

4.1 Experimental setup

4.1.1 Language models

We use unigram language models that assume term independence. Let $p_{MLE}(w|x)$ be the maximum-likelihood estimate of term w with respect to text (collection) x; $p_{MLE}(\cdot|x)$ is referred to as x's unsmoothed language model. Unless otherwise specified, we use a Dirichlet smoothed language model, $p_{Dir}(\cdot|x)$, with parameter μ [40]. To construct relevance models (see Section 4.1.2), we also use the Jelinek-Mercer smoothed language model, $p_{JM}(\cdot|x)$, with smoothing parameter α [40].

To measure similarity between texts x and y, we utilize a previously proposed estimate [18, 19] that is based on the KL-divergence between their induced language models:

$$sim(x,y) \stackrel{def}{=} \exp(-D\left(p_{MLE}(\cdot|x) \;\middle||\; p_{Dir}(\cdot|y)\right)).$$

The merits of this estimate, specifically, with respect to assigning language-model probabilities to long sequences of text (documents in our case), were demonstrated in some previous work [18, 19].

4.1.2 Relevance-feedback methods

To evaluate the effectiveness of the representatives-selection methods, we use two highly effective, yet quite different, relevance-feedback methods that operate in the language modeling framework; namely, the *model-based feedback approach* [39], and *relevance models* [20].

Both methods take as input a query q and a set of relevant documents S, and construct a "topic" language model $p(\cdot|T)$ that is assumed to generate the terms in documents relevant to q. The methods differ by the estimation approach of the topic model as described below. As is common [1], the estimated topic model is then *clipped* by setting $p(w|T)$ to zero for all but the β terms w with the highest $p(w|T)$; normalization is performed to yield a probability distribution $\tilde{p}(\cdot|T)$. We finally derive the *feedback language model* $p(\cdot|S)$ by further anchoring the constructed topic model to the query [39, 1] using interpolation with a free parameter λ: $p(w|S) \stackrel{def}{=} \lambda p_{MLE}(w|q) + (1-\lambda)\tilde{p}(w|T)$. Document d is ranked with respect to the feedback model based on the KL divergence $D\left(p(\cdot|S) \;\middle||\; p_{Dir}(\cdot|d)\right)$.

Estimating a topic model. The model-based feedback approach [39] assumes that each document d in S is generated by a mixture, with free parameter η, of a topic language model $p(\cdot|T)$ with the corpus model $p_{MLE}(\cdot|\mathcal{D})$. To estimate $p(\cdot|T)$, the EM algorithm is used [39].

To estimate a relevance model (RM1) given a set of relevant documents S [21], a centroid of the (Jelinek-Mercer smoothed) language models of documents in S is used: $p(w|T) \stackrel{def}{=} \frac{1}{|S|}\sum_{d\in S} p_{JM}(w|d)$.[7] The resultant feedback model, $p(\cdot|S)$ (a.k.a. RM3 [1]), constructed as described above, is an analog of Rocchio's model [27] in the language-modeling framework [21]. Note that while with RM3 the (query-anchored) centroid of the *selected set* of presumed representatives is used to rank the corpus, in Section 2.1.2 the centroid of *the entire set* of relevant documents ($\mathcal{D}_{rel}$) was used for selecting representatives from $\mathcal{D}_{rel}$ — the Centroid method. We come back to this point in Section 4.2.1.

We note that the model-based feedback approach and the relevance model treat relevant documents as equi-important.

4.1.3 Data, evaluation measures, and parameters

We conducted experiments with the following TREC data:

corpus	# of docs	queries	disk(s)
AP	242,918	51-150	1-3
WSJ	173,252	151-200	1-2
ROBUST	528,155	301-450, 601-700	4-5 (-CR)
WT10g	1,692,096	451-550	WT10g

Titles of TREC topics serve for queries. We applied tokenization, Porter stemming, and stopword removal (using the INQUERY list) using the Lemur toolkit[8], which was used for experiments. The set $\mathcal{D}_{rel}$ of the documents relevant to a query q is that determined by TREC's human annotators ("qrels" files).

The effectiveness of the document-based representatives-selection methods from Section 2.1 is studied when selecting either $k = 1$ or $k = 5$ relevant documents. Each case constitutes a separate experimental setting with respect to tuning of free parameter values. (See the below.) Note that in the single representative case ($k = 1$), we essentially evaluate representativeness by the relative extent to which (language) models of relevant documents in the corpus are similar to that induced from the selected document (and the query), with respect to models of non-relevant documents. For the cluster-based methods (Section 2.2), we set the cluster size (k) to 5, and select the 5 relevant documents in the highest ranked cluster.

The representatives-selection methods are evaluated by the resultant effectiveness of the relevance-feedback methods when fed with the selected documents and the query. An important issue with relevance-feedback evaluation is whether to include the documents used as input in evaluation, or exclude them and use only the *residual* corpus [3]. Since the representatives-selection methods select *different* sets of relevant documents to be used for relevance feedback, the residual-corpus approach suffers from significant problems as previously noted [3]. Thus, we evaluate relevance-feedback performance based on the MAP(@1000) and precision of the top-10 documents (p@10) attained with respect to *all* known relevant documents. Hence, representativeness of selected documents is evaluated based on their effectiveness in helping to find relevant documents — *themselves* and *others* — via relevance-feedback-based search performed over the corpus. Statistically significant performance differences are determined using the two-tailed paired t-test at a 95% confidence level [29, 31].

[7] When no relevance feedback is available, relevance model construction is based on a pseudo feedback approach [20].
[8] www.lemurproject.org

	AP				WSJ				ROBUST				WT10g			
	$k=1$		$k=5$		$k=1$		$k=5$		$k=1$		$k=5$		$k=1$		$k=5$	
	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10
	Model-based feedback															
Random	32.9	60.1	40.7	77.7	42.4	58.0	50.7	68.6	35.8	56.5	42.4	63.9	30.8	42.9	39.3	51.5
QuerySim	33.6	61.4	40.8	76.4	40.8	56.8	51.5	72.4	32.8^{r}	53.9	41.7	66.6	32.6	44.4	39.7	55.1
Length	33.7	62.5	**45.9**$^{r}_{s}$	79.6	41.1	56.6	**55.2**$^{r}_{s}$	75.0^{r}	37.7$^{r}_{s}$	59.2$^{r}_{s}$	**50.3**$^{r}_{s}$	**75.1**$^{r}_{s}$	31.0	43.9	**44.8**$^{r}_{s}$	62.2$^{r}_{s}$
Entropy	33.6	63.1	44.7$^{r}_{s}$	78.5	41.6	57.8	**55.2**$^{r}_{s}$	74.8^{r}	37.3$^{r}_{s}$	58.2$_{s}$	48.5$^{r}_{s}$	73.7$^{r}_{s}$	31.9	44.4	41.9	57.7^{r}
Clarity	32.9	60.2	41.0	75.0	39.9	55.4	54.8$^{r}_{s}$	**76.4**$^{r}_{s}$	37.6$^{r}_{s}$	60.2$^{r}_{s}$	47.1$^{r}_{s}$	71.9$^{r}_{s}$	30.2	43.6	39.7	55.1
Centroid	40.6$^{r}_{s}$	**76.2**$^{r}_{s}$	43.0^{r}	**80.3**	**49.6**$^{r}_{s}$	**69.6**$^{r}_{s}$	54.3$^{r}_{s}$	73.8^{r}	42.6$^{r}_{s}$	65.9$^{r}_{s}$	45.2$^{r}_{s}$	72.3$^{r}_{s}$	37.9$^{r}_{s}$	54.9$^{r}_{s}$	43.7$^{r}_{s}$	**62.9**$^{r}_{s}$
WInDeg	36.8$^{r}_{s}$	70.6$^{r}_{s}$	40.2	77.9	47.6$^{r}_{s}$	67.2$^{r}_{s}$	52.1	70.6	41.6$^{r}_{s}$	64.9$^{r}_{s}$	42.3	66.5	37.4$^{r}_{s}$	55.0$^{r}_{s}$	40.4	58.8$^{r}_{s}$
UInDeg	38.4$^{r}_{s}$	71.6$^{r}_{s}$	43.0^{r}	79.3	47.0$^{r}_{s}$	66.2$^{r}_{s}$	53.6	71.6	41.2$^{r}_{s}$	64.6$^{r}_{s}$	46.1$^{r}_{s}$	70.4$^{r}_{s}$	37.9$^{r}_{s}$	52.9$^{r}_{s}$	42.9$^{r}_{s}$	59.7$^{r}_{s}$
WPR	38.2$^{r}_{s}$	72.5$^{r}_{s}$	42.4	79.1	48.4$^{r}_{s}$	67.8$^{r}_{s}$	53.1	71.2	42.1$^{r}_{s}$	65.0$^{r}_{s}$	45.9$^{r}_{s}$	70.1$^{r}_{s}$	38.6$^{r}_{s}$	54.7$^{r}_{s}$	42.9$^{r}_{s}$	59.7$^{r}_{s}$
UPR	38.1$^{r}_{s}$	70.5$^{r}_{s}$	42.3	78.2	48.2$^{r}_{s}$	68.0$^{r}_{s}$	52.8	71.4	41.6$^{r}_{s}$	64.9$^{r}_{s}$	45.5$^{r}_{s}$	69.5$^{r}_{s}$	37.7$^{r}_{s}$	52.6$^{r}_{s}$	42.5$^{r}_{s}$	59.0$^{r}_{s}$
MMR	33.6	61.4	42.3	73.5	40.8	56.8	52.6	67.4	32.8^{r}	53.9	43.9$_{s}$	64.9	32.6	44.4	38.2	50.8$_{s}$
LTR	**40.8**$^{r}_{s}$	74.7$^{r}_{s}$	43.3$^{r}_{s}$	80.2$_{s}$	47.8$^{r}_{s}$	66.8$^{r}_{s}$	53.9^{r}	71.4	**42.9**$^{r}_{s}$	**66.0**$^{r}_{s}$	44.7$^{r}_{s}$	70.9$^{r}_{s}$	**38.8**$^{r}_{s}$	**56.3**$^{r}_{s}$	43.7$^{r}_{s}$	60.7$^{r}_{s}$
	Relevance model															
Random	30.9	58.0	43.9	78.9	42.5	58.2	52.7	71.8	35.8	56.6	50.3	74.9	31.1	42.4	**45.8**	63.3
QuerySim	33.6	62.9	41.1^{r}	77.4	40.8	56.0	51.8	72.8	32.5^{r}	53.7^{r}	44.6^{r}	70.6^{r}	32.5	44.6	43.6	60.2
Length	31.3	59.6	**46.3**$^{r}_{s}$	79.8	41.3	57.0	54.4	73.6	37.7$^{r}_{s}$	59.3$^{r}_{s}$	51.2$_{s}$	75.1$_{s}$	31.0	44.3	45.7	61.2
Entropy	30.9	58.1	45.8$_{s}$	78.9	41.6	58.0	54.6$_{s}$	72.6	37.3$^{r}_{s}$	58.2$_{s}$	51.0$_{s}$	74.8$_{s}$	31.9	44.6	45.1	62.4
Clarity	30.5	55.9$_{s}$	43.9$_{s}$	78.5	40.0	56.2	54.3$_{s}$	**76.6**^{r}	37.8$^{r}_{s}$	60.5$^{r}_{s}$	51.2$_{s}$	76.2$_{s}$	30.7	43.8	44.4	62.8
Centroid	39.8$^{r}_{s}$	**76.3**$^{r}_{s}$	44.1$_{s}$	**81.3**$_{s}$	**49.1**$^{r}_{s}$	**66.8**$^{r}_{s}$	54.0	74.8	42.2$^{r}_{s}$	65.4$^{r}_{s}$	47.2$^{r}_{s}$	74.8$_{s}$	38.0$^{r}_{s}$	55.3$^{r}_{s}$	43.2^{r}	63.5
WInDeg	35.5^{r}	70.1$^{r}_{s}$	41.9	79.4	46.9$^{r}_{s}$	65.4$^{r}_{s}$	52.5	74.2	41.3$^{r}_{s}$	64.5$^{r}_{s}$	46.8$^{r}_{s}$	73.0$_{s}$	37.4$^{r}_{s}$	54.3$^{r}_{s}$	39.4$^{r}_{s}$	59.2^{r}
UInDeg	37.6$^{r}_{s}$	71.7$^{r}_{s}$	44.3$_{s}$	80.5	46.3$^{r}_{s}$	64.0$^{r}_{s}$	54.7	73.4	41.0$^{r}_{s}$	64.2$^{r}_{s}$	49.0$_{s}$	74.1$_{s}$	37.8$^{r}_{s}$	52.7$^{r}_{s}$	45.6	**63.6**$_{s}$
WPR	37.3$^{r}_{s}$	71.9$^{r}_{s}$	44.1$_{s}$	79.5	47.7$^{r}_{s}$	66.0$^{r}_{s}$	53.2	73.4	41.6$^{r}_{s}$	64.7$^{r}_{s}$	48.6$^{r}_{s}$	73.9$_{s}$	38.3$^{r}_{s}$	54.7$^{r}_{s}$	45.3	63.2
UPR	37.1$^{r}_{s}$	70.3$^{r}_{s}$	43.9$_{s}$	79.4	47.5$^{r}_{s}$	66.4$^{r}_{s}$	53.2	73.4	41.4$^{r}_{s}$	64.4$^{r}_{s}$	48.2$^{r}_{s}$	73.3$_{s}$	37.8$^{r}_{s}$	52.5$^{r}_{s}$	45.2	63.4$_{s}$
MMR	33.6	62.9	44.5$_{s}$	76.9	40.8	56.0	**55.1**$_{s}$	74.2	32.5^{r}	53.7^{r}	**52.1**$^{r}_{s}$	**77.0**$^{r}_{s}$	32.5	44.6	45.2	60.9
LTR	**40.8**$^{r}_{s}$	74.4$^{r}_{s}$	45.1$_{s}$	81.2$_{s}$	47.7$^{r}_{s}$	65.2$^{r}_{s}$	53.8	75.0	**42.6**$^{r}_{s}$	**66.0**$^{r}_{s}$	47.8$^{r}_{s}$	74.8$_{s}$	**38.7**$^{r}_{s}$	**55.8**$^{r}_{s}$	44.7	63.2

Table 1: Evaluation of document-based methods for selecting k representative relevant documents. (See Section 2.1.) Best result in a column is boldfaced. Statistically-significant differences with the Random and QuerySim methods are marked with 'r' and 's', respectively.

Parameter values. Our goal is to focus on the underlying principles of the representatives-selection methods, rather than engage in excessive parameter tuning. Therefore, we have taken the following experimental-design decisions.

The document language model Dirichlet-smoothing parameter (μ) is set to 1000 in all methods following previous recommendations [40]. The free parameters of the graph-based methods, δ and ν, are set to 5 and 0.8, respectively, following limited experimentation with $\delta \in \{5, 10\}$ and $\nu \in \{0.2, 0.8\}$ and previous recommendations [18]. The free parameter of the MMR method, ψ, is set to 0.1 following experiments with $\psi \in \{0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9\}$. The SVM^{rank} package [15], used by the learning-to-rank methods, is employed with default parameter values. For learning/testing we use ten fold cross validation performed over the queries for a corpus; each fold contains a consecutive set of queries, wherein queries are ordered by their IDs.

The free parameters of the relevance feedback methods are set for *all* representatives-selection approaches to values that yield optimized MAP performance for the QuerySim approach, which serves as a reference comparison to all methods. The search ranges for parameter values are: $\alpha \in \{0, 0.1, 0.3, \ldots, 0.9, 1\}$, $\beta \in \{5, 10, 25, 50, 75, 100\}$, $\lambda \in \{0, 0.1, 0.3, \ldots, 0.9, 1\}$, and $\eta \in \{0.1, 0.3, \ldots, 0.9\}$.

4.2 Experimental results

4.2.1 Document-based methods

In Table 1 we present the performance of the document-based representatives-selection methods from Section 2.1. As can be seen, selecting documents based on their query-similarity (QuerySim) can often be less effective than random (Random) selection. Henceforth, we use these two methods as reference comparisons to all other approaches.

We can see in Table 1 that for selecting $k = 5$ representatives, short documents (Length), documents with term distribution concentrated around a relatively small number of terms (Entropy), and documents with models distant from that of the corpus (Clarity), are all good choices, with Length being the most effective approach among the three[9]. Indeed, note that several of the boldface marks in the table, which indicate the best performance per corpus, evaluation measure, and k, appear in lines corresponding to these three methods when $k = 5$. Furthermore, Length, Entropy and Clarity post performance for $k = 5$ that is in most relevant comparisons better than that of Random and QuerySim; these improvements are often also statistically significant.

However, Length, Entropy and Clarity are much less effective for selecting a single representative ($k = 1$), specifically, with respect to methods that utilize inter-document relationships, and which we discuss below. This finding is not surprising. A case in point, when using 5 relevant documents, which is considered a "decent" sample [11], the relevance-feedback methods essentially "emphasize" the commonalities between these documents; taking care, in addi-

[9]Experimental results — actual numbers are omitted due to space considerations — show that (i) selecting short documents is much preferable to selecting long documents, (ii) selecting documents with low entropy is much preferable to selecting documents with high entropy, and (iii) selecting documents with high clarity is preferable to selecting documents with low clarity. These results support the hypotheses stated in Section 2.1, based on which Length, Entropy and Clarity were devised.

tion, that the documents are somewhat focused (as estimated by Length, Entropy, and Clarity), improves the representativeness of the constructed feedback model. On the other hand, selecting a single document, as focused as it might be, independently of other documents, is less likely to result in a representative feedback model due to non query-related aspects potentially manifested in the document.

We now turn to explore methods utilizing inter-document relationships. It is evident in Table 1 that similarity to the centroid of the relevant-documents set (Centroid) is a highly effective representatives-selection approach, which in many cases is superior to the other methods. Specifically, Centroid almost always outperforms — often to a a substantial and statistically-significant degree — Random and QuerySim.

At first glance, the performance of Centroid with the relevance model could be viewed as somewhat biased. That is, the Centroid method selects representatives based on similarity to the centroid of the *relevant-documents* set, while the relevance model uses a (query-anchored) centroid of the *selected* representatives to rank the corpus. If these two centroids are quite close to each other, then the Centroid method could potentially be regarded as having "advantage" over other methods. However, a closer look at Table 1 reveals that there are many cases wherein the Centroid is outperformed by other methods when using the relevance model — e.g., by Length for many comparisons with $k = 5$, and by Random (when MAP is considered) over WT10g.

Table 1 shows that the graph-based methods, which utilize inter-document similarities, WInDeg, WPR, UInDeg, and UPR, are also quite effective. Specifically, all four approaches outperform Random and QuerySim in almost all relevant comparisons; often, to a statistically significant degree. Among the four, UInDeg, which counts for a document the number of documents it is among the nearest-neighbors of, and WPR, which is a weighted version of PageRank employed over the similarity-based graph, are the most effective. Thus, we see that documents centrally located within the similarity-based graphs tend to be good representatives. Moreover, as noted above, for selecting a single representative, utilizing inter-document relationships, as in the graph-based and Centroid methods, yields performance substantially better than that of considering documents independently (QuerySim, Length, Entropy, and Clarity).

The effectiveness of the graph-based and Centroid methods in finding a single representative provides an interesting perspective on the cluster hypothesis [35]. Recall that for the single representative case, the relevance feedback methods essentially search the corpus for documents with models similar to that of the selected document (anchored to the query). Thus, the effectiveness just mentioned implies that relevant documents centrally located in the similarity space *of relevant documents* are closer to relevant documents than to non-relevant documents; non centrally-located relevant documents might manifest these relative similarities to a lesser extent. Thus, while the cluster hypothesis implies that relevant documents are similar to each other, the observation just stated could be viewed as its refinement.

We next turn to examine the MMR approach, which looks for a diversified set of representatives. Evidently, MMR is effective with the relevance model, but much less effective with the model-based feedback method. In the latter case it is often outperformed by Random and QuerySim. (Recall that for $k = 1$ MMR amounts to QuerySim.)

Finally, we turn to the learning-to-rank approach, LTR, which integrates all methods from above (except for MMR and Random). Clearly, LTR is highly effective. It almost always outperforms Random and QuerySim; often to a substantial and statistically significant degree. We note that the Centroid is assigned with the highest weight by the learner, and that this weight is much higher than that assigned to other methods. Yet, the overall superiority of LTR to Centroid when using the model-based feedback method attests that other approaches can yield further improvements on top of that posted by Centroid. Among those, as indicated by the weights the learner assigns, are QuerySim and WInDeg.

Summary. All in all, we see that methods that look for documents centrally located in the similarity space are highly effective representatives-selection approaches, especially for selecting a single representative; and, that integrating those with other methods can further improve performance.

4.2.2 Cluster-based methods

In Table 2 we present the performance of the cluster-based methods for selecting $k = 5$ representative documents — i.e., those in the highest ranked cluster. The first four rows present performance numbers for reference comparisons. The first, **Opt. Clust.**, is manually selecting a cluster for each query so that the MAP performance of the relevance-feedback method when fed with the cluster's constituent documents and the query is the highest with respect to all other clusters. Hence, the performance of Opt. Clust. serves as an upper bound for that of all cluster-based methods in the table. The second reference comparison, **clust-Random**, selects a cluster randomly. The third and fourth, **doc-Random** and **doc-QuerySim**, are the Random and QuerySim methods from Table 1, which select relevant *documents* (rather than clusters) randomly, and based on document-query similarities, respectively.

Our first observation based on Table 2 is that there are highly representative clusters, as the performance numbers for Opt. Clust. attest. Finding these clusters results in performance that is by far better than that of all other methods considered in this paper.

We can also see in Table 2 that the cluster-based methods (clust-QuerySim,..., clust-LTR) yield performance that is almost always better — often to a substantial extent — than that of selecting a random cluster (clust-Random). (Many of these performance improvements are also statistically-significant; these were not marked in the table to avoid cluttering.) However, the methods that are based on the mean document score in the cluster (clust-QuerySim,..., clust-UPR), are (i) consistently less effective than their counterparts in Table 1 that use the scores to directly rank documents, and (ii) often less effective than the document-based baselines, doc-Random and doc-QuerySim; in many cases, to a statistically-significant degree. The latter also holds for the cluster-density-based methods, BaseDense and Avg-Dense. Thus, we see that identifying representative clusters is a hard challenge.

Another observation that we make based on Table 2 is that Centroid is the most effective document-based scoring method in terms of the resultant cluster-based performance (clust-Centroid). Furthermore, clust-Centroid is the method that is assigned with the highest weight by the learner in

	AP		WSJ		ROBUST		WT10g	
	MAP	p@10	MAP	p@10	MAP	p@10	MAP	p@10
	Model-based feedback							
Opt. Clust.	51.7$^{r}_{s}$	88.3$^{r}_{s}$	61.8$^{r}_{s}$	80.8$^{r}_{s}$	55.2$^{r}_{s}$	81.5$^{r}_{s}$	51.8$^{r}_{s}$	69.3$^{r}_{s}$
clust-Random	33.5$^{r}_{s}$	68.7$^{r}_{s}$	43.9$^{r}_{s}$	64.8$_{s}$	37.4$^{r}_{s}$	61.2$_{s}$	36.6$_{s}$	52.7
doc-Random	40.7	77.7	50.7	68.6	42.4	63.9	39.3	51.5
doc-QuerySim	40.8	76.4	51.5	72.4	41.7	66.6	39.7	55.1
clust-QuerySim	37.3$^{r}_{s}$	73.5	49.3$_{s}$	67.4$_{s}$	40.0$^{r}_{s}$	65.3	40.0	54.4
clust-Length	34.2$^{r}_{s}$	69.6$^{r}_{s}$	50.1	66.4$_{s}$	43.1	67.3	**42.3**$^{r}_{s}$	60.0$^{r}_{s}$
clust-Entropy	34.6$^{r}_{s}$	68.7$^{r}_{s}$	47.5	64.8$_{s}$	38.2$^{r}_{s}$	57.9$^{r}_{s}$	39.1	53.0
clust-Clarity	34.9$^{r}_{s}$	68.1$^{r}_{s}$	48.6	65.4$_{s}$	38.9$^{r}_{s}$	59.1$^{r}_{s}$	35.3$^{r}_{s}$	48.4$_{s}$
clust-Centroid	40.5	77.5	50.8	71.0	43.4$_{s}$	70.2$^{r}_{s}$	42.0$^{r}_{s}$	**61.5**$^{r}_{s}$
clust-WInDeg	35.9$^{r}_{s}$	75.2	48.7	68.6	39.0$^{r}_{s}$	63.9	39.3	56.9^{r}
clust-UInDeg	37.0$^{r}_{s}$	74.9	49.4	68.0	42.3	67.6^{r}	40.6	55.4
clust-WPR	37.0$^{r}_{s}$	74.2	49.4	69.0	42.4	68.6^{r}	41.3	57.7^{r}
clust-UPR	37.1$^{r}_{s}$	74.7	49.6	68.8	42.4	68.4^{r}	41.6$_{s}$	58.6$^{r}_{s}$
clust-BaseDense	37.1$^{r}_{s}$	71.7$^{r}_{s}$	49.0	69.4	40.0^{r}	62.3$_{s}$	39.9	54.5
clust-AvgDense	35.3$^{r}_{s}$	66.5$^{r}_{s}$	49.1	67.0	41.3	64.2	38.9	52.6
clust-LTR	**42.1**	**80.0**$_{s}$	**52.2**	**71.4**	**44.5**$^{r}_{s}$	**70.8**$^{r}_{s}$	42.0$^{r}_{s}$	58.9$^{r}_{s}$
	Relevance model							
Opt. Clust.	52.7$^{r}_{s}$	88.6$^{r}_{s}$	62.0$^{r}_{s}$	80.4$^{r}_{s}$	58.2$^{r}_{s}$	83.7$^{r}_{s}$	54.7$^{r}_{s}$	72.6$^{r}_{s}$
clust-Random	33.7$^{r}_{s}$	68.0$^{r}_{s}$	43.7$^{r}_{s}$	64.2^{r}	41.4$^{r}_{s}$	68.0$^{r}_{s}$	38.6$^{r}_{s}$	57.7^{r}
doc-Random	43.9	78.9	52.7	71.8	50.3	74.9	45.8	63.3
doc-QuerySim	41.1	77.4	51.8	72.8	44.6	70.6	43.6	60.2
clust-QuerySim	38.2$^{r}_{s}$	74.9^{r}	49.6$_{s}$	70.6	42.7$^{r}_{s}$	68.7$^{r}_{s}$	43.6	61.3
clust-Length	36.8$^{r}_{s}$	72.5$^{r}_{s}$	50.3	69.2	46.3^{r}	72.5	42.8^{r}	61.1
clust-Entropy	37.4$^{r}_{s}$	73.7^{r}	49.3	68.2	46.2^{r}	71.4^{r}	42.7^{r}	60.1
clust-Clarity	37.8$^{r}_{s}$	72.3$^{r}_{s}$	51.4	**73.4**	46.2^{r}	72.0^{r}	42.1^{r}	59.8
clust-Centroid	41.9^{r}	78.2	50.7	72.6	45.4^{r}	72.5^{r}	41.0$^{r}_{s}$	61.6
clust-WInDeg	37.2$^{r}_{s}$	76.8	48.8^{r}	70.6	42.6$^{r}_{s}$	68.9^{r}	38.3$^{r}_{s}$	58.4^{r}
clust-UInDeg	38.2$^{r}_{s}$	76.8	49.5^{r}	71.2	44.8^{r}	71.0^{r}	43.1^{r}	61.2
clust-WPR	38.3^{r}	76.4	49.1^{r}	72.2	44.5^{r}	70.6^{r}	42.0^{r}	61.2
clust-UPR	38.3^{r}	76.5	49.6^{r}	72.4	44.5^{r}	70.6^{r}	43.0^{r}	**62.3**
clust-BaseDense	37.8$^{r}_{s}$	73.1$^{r}_{s}$	49.6^{r}	70.6	43.9^{r}	69.2^{r}	42.4^{r}	58.2^{r}
clust-AvgDense	36.8$^{r}_{s}$	68.4$^{r}_{s}$	50.0	70.4	46.0^{r}	71.2^{r}	43.3^{r}	60.1
clust-LTR	**43.3**$_{s}$	**80.3**	**51.8**	73.0	**46.5**$^{r}_{s}$	**73.9**$_{s}$	**43.7**	62.0

Table 2: Evaluation of cluster-based methods (clust-X) for selecting 5 representative relevant documents. (Refer to Section 2.2.) The document-based (doc-)Random and (doc-)QuerySim methods from Table 1 are presented for reference; 'r', and 's' mark statistically-significant differences with these reference comparisons, respectively. Boldface marks the best *cluster-based* performance in a column.

the learning-to-rank approach of clusters (clust-LTR). These findings echo those from Table 1.

Clearly, the most effective cluster-based selection method in Table 2 is the learning-to-rank (clust-LTR) approach. Specifically, clust-LTR posts performance that is often better — sometimes to a statistically-significant degree — than that of the document-based reference comparisons (doc-Random and doc-QuerySim). Although clust-Centroid is the method assigned with the highest weight by the learner, as noted above, clust-LTR consistently outperforms clust-Centroid. This finding attests to the (somewhat) complementary nature of the methods integrated by clust-LTR. Specifically, clust-QuerySim and clust-AvgDense are assigned with relatively high weights by the learner, albeit much lower than that of clust-Centroid.

4.2.3 Poison pills

Previous work showed that there are queries for which some relevant documents are quite ineffective for relevance-feedback-based retrieval regardless of the retrieval method used [37, 33]. Some of these documents, termed *poison pills*, when used alone (with the query), yield performance that is even worse than that of not utilizing relevance feedback at all [33]. However, characterizing and automatically identifying such documents is an open question [33].

	AP	WSJ	ROBUST	WT10g
	Model-based feedback			
QL	22.2	31.4	25.0	20.6
true-worst	9.3	23.2	18.7	15.2
true-best	48.3	55.5	50.3	47.2
Random	32.9	42.4	35.8	30.8
QuerySim-worst	24.5	37.3	34.3	29.6
QuerySim-best	33.6	40.8	32.8	32.6
Centroid-worst	17.5	33.1	26.4	21.3
Centroid-best	40.6	49.6	42.6	37.9
	Relevance model			
QL	22.2	31.4	25.0	20.6
true-worst	5.4	24.9	19.1	15.9
true-best	49.1	55.2	49.8	47.0
Random	30.9	42.5	35.8	31.1
QuerySim-worst	19.3	38.4	34.3	30.6
QuerySim-best	33.6	40.8	32.5	32.5
Centroid-worst	12.6	33.8	26.9	22.5
Centroid-best	39.8	49.1	42.2	38.0

Table 3: The relevance-feedback-based MAP performance of using a *single* document considered as the worst (best) *representative* by a representatives-selection method. The performance of using the relevant document that actually yields the worst (best) performance — i.e., the "true-worst(best)" representative *per* relevance feedback method, and that of a standard query-likelihood (QL) retrieval model that does not utilize relevance feedback, is presented for reference.

We therefore turn to explore whether relevant documents considered as "bad" representatives by our methods — i.e., that are assigned with low representativeness values — are indeed ineffective for relevance-feedback-based retrieval. More generally, we further study the effectiveness of our document-based methods (refer to Sections 2.1 and 4.2.1) in differentiating "good" from "bad" representatives.

Table 3 presents the MAP performance of relevance-feedback-based retrieval when using the worst and best representatives identified by some of our methods — i.e., the documents that are assigned with the lowest and highest representativeness values, respectively. We also present the performance of the "true-worst" and "true-best" representatives *per* relevance feedback method; that is, the performance of using the documents that actually yield the worst and best performance, respectively. Thus, while "true-worst" and "true-best" are relevance-feedback-method specific, our methods define representativeness *regardless* of the relevance-feedback method used. In addition, the performance of the query likelihood (**QL**) retrieval model [32], which scores document d in the corpus by $sim(q, d)$, and which does not utilize relevance feedback, is presented for reference.

We can see in Table 3 that using "true-worst" yields performance that is substantially worse than that of not using relevance feedback at all (the QL method); and, using "true-best" yields much better performance than that of QL. These findings further attest to the significant performance impact of the relevant documents used for feedback.

	AP	WSJ	ROBUST	WT10g
	Model-based feedback			
Ground truth	85.9	76.0	70.7	66.0
Random	19.2	18.0	15.3	23.0
QuerySim	40.4	34.0	19.7	23.0
Length	33.3	30.0	32.5	35.0
Centroid	61.6	44.0	40.2	42.0
	Relevance model			
Ground truth	87.9	72.0	70.3	66.0
Random	30.3	16.0	15.3	20.0
QuerySim	54.5	20.0	19.7	19.0
Length	46.5	22.0	31.3	30.0
Centroid	69.7	32.0	39.4	38.0

Table 4: The percentage of queries for which the document considered as the worst representative by a method is a "poison pill" [33] in terms of MAP performance. "Ground truth" is the percentage of queries for which a poison pill exists.

We can also see in Table 3 that random selection of a relevant document yields performance that is much better than that of QL, but much worse than that of "true-best".

Another observation that we make based on Table 3 is that the QuerySim method, which utilizes document-query similarities, does not effectively differentiate good from bad representatives. That is, the difference in performance between using the worst and best identified representative is quite small (except for the AP case) when compared, for example, with that posted by the Centroid method. In fact, for the ROBUST corpus, the "best" identified representative of QuerySim yields performance that is worse than that of the "worst" identified representative.

Table 3 also shows that the Centroid method is quite effective in differentiating good from bad representatives. Indeed, the performance difference between using the worst and best representative is quite large. Furthermore, the performance of using the best Centroid-based representative is much better than that of random document selection; and, using the worst Centroid-based representative yields performance that is much worse than that of random selection. Yet, the worst and best representatives identified by Centroid still yield performance quite different than that of "true-worst" and "true-best", respectively.

In further exploration, we present in Table 4 the percentage of queries for which our methods find a "poison pill"; that is, the percentage of queries for which using the document considered as the worst representative (along with the query) yields MAP performance that is worse than that of QL. Table 4 also presents the "ground truth" numbers — the percentage of queries for which a poison pill exists.

As can be seen in Table 4, the worst representatives identified by our methods are much more often poison pills than randomly selected relevant documents. Furthermore, in almost all cases, for more than half of the queries for which a poison pill exists, the least representative document as determined by the Centroid method tends to be a poison pill. These findings further show that relevant documents considered as bad representatives by our methods tend to be quite ineffective for relevance-feedback-based retrieval.

	NDCG@3	NDCG@5	NDCG@10
Random	28.4	35.8	48.1
MBF-true	29.6^r	37.5^r	50.3^r
RelModel-true	29.2	36.8	49.7^r
QuerySim	**31.9**^{r}	**39.6**^{r}	**52.5**^{r}
Length	28.1	36.0	48.6
Entropy	27.8	35.4	48.0
Clarity	28.6	36.0	48.7
Centroid	29.7^r	37.4^r	49.9^r
WInDeg	29.2^r	36.9^r	49.4^r
UInDeg	29.9	37.8	50.4^r
WPR	30.1^r	37.9^r	50.6^r
UPR	29.8^r	37.7^r	50.4^r
MMR	30.0^r	37.2^r	49.9^r

Table 5: The connection between representativeness and gradual relevance judgments. We measure NDCG@k for the WT10g corpus when ranking relevant documents based on their representativeness and using level-1 ("relevant") and level-2 ("highly relevant") relevance judgments. MBF-true and RelModel-true denote a ranking induced by the "true" representativeness level of a document *with respect* to the model-based feedback and relevance model approaches, respectively. The best result in a column is boldfaced; 'r' marks statistically-significant difference with Random ranking.

4.2.4 Gradual relevance judgments and representativeness

The next question we turn to explore is whether representative relevant documents would be considered by human judges as "more relevant" than non-representative relevant documents. To that end, we use the WT10g corpus for which gradual relevance judgments are available; specifically, a relevant document is marked with relevance-level "1" ("relevant") or "2" ("highly relevant"). We rank the relevant documents in $\mathcal{D}_{rel}$ by the representativeness values assigned by the different document-based methods, and compute NDCG [14] at various cutoffs using the two relevance levels. We present for reference the NDCG of a ranking induced by the "true" representativeness level of a document as determined *with respect* to the model-based feedback approach ("MBF-true") and *with respect* to the relevance model approach ("RelModel-true") — i.e., with respect to the resultant relevance-feedback-based performance when using the document with the query.

We can see in Table 5 that most of our representatives-selection methods yield ranking that is superior to random ranking; many of the improvements posted over random ranking by methods that utilize inter-document similarities (e.g., the Centroid and the graph-based methods) are also statistically significant. Furthermore, some of our methods post performance that is superior to that of a ranking based on the "true" representativeness as determined *with respect* to a relevance feedback method. These findings attest to the connection between the notion of representativeness employed by our methods, which is relevance-feedback-method agnostic, and TREC's gradual relevance judgments.

However, we also see in Table 5 that the QuerySim method, which was among the worst considered above for finding representative documents, is the most effective for differentiating between "relevant" and "highly relevant" documents. Thus, while relevant documents that exhibit high surface-

level query similarity tend to be deemed "highly relevant" by TREC's annotators, these documents, as shown above, are not highly effective in helping to find other relevant documents using relevance-feedback-based retrieval.

5. CONCLUSION

We have addressed the question of what makes some relevant documents more effective than others for use in relevance-feedback-based retrieval. Specifically, we presented a suite of methods for estimating the *representativeness* of relevant documents; the level of representativeness is measured by the resultant performance of *some* relevance feedback method when fed with the documents and the query, and employed over the corpus. We showed that documents centrally located within the similarity space of the relevant-documents set are good representatives. Furthermore, we showed that there are highly representative *clusters* of similar relevant documents, and proposed methods for ranking clusters based on their presumed representativeness. In addition, we showed that documents considered as "bad" representatives by our methods often yield ineffective relevance-feedback-based retrieval performance. Finally, we demonstrated the connection between representativeness and TREC's gradual relevance judgments.

Acknowledgments We thank the reviewers for their comments. This paper is based upon work supported in part by Israel's Science Foundation under grant no. 890015, by G. S. Elkin research fund at the Technion, by Google's and IBM's faculty research awards, and by IBM's SUR award. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsoring institutions.

6. REFERENCES

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade. UMASS at TREC 2004 — novelty and hard. In *Proceedings of TREC-13*, pages 715–725, 2004.

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

[3] C. Buckley and S. Robertson. Relevance feedback track overview: TREC 2008. In *Proceedings of TREC-17*, 2008.

[4] S. Büttcher, C. L. A. Clarke, P. C. K. Yeung, and I. Soboroff. Reliable information retrieval evaluation with incomplete and biased judgements. In *SIGIR*, pages 63–70, 2007.

[5] J. G. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of SIGIR*, pages 335–336, 1998.

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

[7] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *Proceedings of SIGIR*, pages 299–306, 2002.

[8] F. Diaz. Regularizing ad hoc retrieval scores. In *Proceedings of CIKM*, pages 672–679, 2005.

[9] M. D. Dunlop. The effect of accessing nonmatching documents on relevance feedback. *ACM Transactions on Information Systems*, 15(2):137–153, 1997.

[10] 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.

[11] D. Harman. Relevance feedback revisited. In *Proceedings of SIGIR*, pages 1–10, 1992.

[12] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of SIGIR*, pages 76–84, 1996.

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

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

[15] T. Joachims. Training linear SVMs in linear time. In *Proceedings of KDD*, pages 217–226, 2006.

[16] O. Kurland. *Inter-document similarities, language models, and ad hoc retrieval*. PhD thesis, Cornell University, 2006.

[17] O. Kurland. The opposite of smoothing: A language model approach to ranking query-specific document clusters. In *Proceedings of SIGIR*, pages 171–178, 2008.

[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 and L. Lee. Respect my authority! HITS without hyperlinks utilizing cluster-based language models. In *Proceedings of SIGIR*, pages 83–90, 2006.

[20] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of SIGIR*, pages 120–127, 2001.

[21] V. Lavrenko and W. B. Croft. Relevance models in information retrieval. In Croft and Lafferty [6], pages 11–56.

[22] T.-Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3), 2009.

[23] X. Liu and W. B. Croft. Experiments on retrieval of optimal clusters. Technical Report IR-478, Center for Intelligent Information Retrieval (CIIR), University of Massachusetts, 2006.

[24] X. Liu and W. B. Croft. Evaluating text representations for retrieval of the best group of documents. In *Proceedings of ECIR*, pages 454–462, 2008.

[25] Y. Lv and C. Zhai. Adaptive relevance feedback in information retrieval. In *Proceedings of CIKM*, pages 255–264, 2009.

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

[27] J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, 1971.

[28] I. Ruthven and M. Lalmas. A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review*, 18(2):95–145, 2003.

[29] M. Sanderson and J. Zobel. Information retrieval system evaluation: Effort, sensitivity and reliability. In *Proceedings of SIGIR*, pages 162–169, 2005.

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

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

[32] F. Song and W. B. Croft. A general language model for information retrieval (poster abstract). In *Proceedings of SIGIR*, pages 279–280, 1999.

[33] E. L. Terra and R. Warren. Poison pills: harmful relevant documents in feedback. In *Proceedgins of CIKM*, pages 319–320, 2005.

[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] E. M. Voorhees and L. P. Buckland, editors. *The Eighteenth Text REtrieval Conference (TREC-18)*. NIST, 2009.

[37] R. H. Warren and T. Liu. A review of relevance feedback experiments at the 2003 reliable information access RIA workshop. In *Proceedings of SIGIR*, pages 570–571, 2004.

[38] C. Zhai, W. W. Cohen, and J. D. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of SIGIR*, pages 10–17, 2003.

[39] C. Zhai and J. D. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of CIKM*, pages 403–410, 2001.

[40] 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.