Incorporating Diversity and Density in Active Learning for Relevance Feedback

Zuobing Xu, Ram Akella, and Yi Zhang

University of California, Santa Cruz, CA, USA, 95064

Abstract. Relevance feedback, which uses the terms in relevant documents to enrich the user's initial query, is an effective method for improving retrieval performance. An associated key research problem is the following: Which documents to present to the user so that the user's feedback on the documents can significantly impact relevance feedback performance. This paper views this as an active learning problem and proposes a new algorithm which can efficiently maximize the learning benefits of relevance feedback. This algorithm chooses a set of feedback documents based on relevancy, document diversity and document density. Experimental results show a statistically significant and appreciable improvement in the performance of our new approach over the existing active feedback methods.

1 Introduction

Information retrieval has traditionally been based on retrieving documents which match user's query in content. It is well known that the original query formulation does not always reflect the user's intent. In other words, merely matching words (or "terms") in the original query and the document may not be an effective approach, as the word overlap alone may not capture the semantic intent of a query. In particular, without detailed knowledge of the collection make-up, and of the retrieval environment, most users find it difficult to formulate information queries that are well designed for retrieval purposes. This suggests that the first retrieval operation can be conducted with a tentative initial query, which retrieves a few useful documents for user to evaluate their relevance. Based on the relevance evaluation and the initial query, we construct a new improved query to retrieve more relevant documents in subsequent operations.

The above retrieval process is well known as relevance feedback process [1,2]. There are two major problems while using relevance feedback framework. First, how to select first set of documents to be presented to the user for feedback. Second, how to effectively utilize the relevant feedback information to reformulate the query. Much of the previous research on relevance feedback focuses on the second problem of feedback query updating for a given set of feedback documents by choosing important topic related terms from the relevant documents and expanding the original query based on the chosen terms.

However, how to choose a good set of documents is not well studied in the information retrieval community, although an effective approach has much potential to further enhance retrieval performance. Most of the earlier relevance

G. Amati, C. Carpineto, and G. Romano (Eds.): ECIR 2007, LNCS 4425, pp. 246–257, 2007.

feedback systems usually ignore the first problem and choose top ranked documents for feedback. This ignores many important factors that affect the learning results. Recently, Shen and Zhai [3] presented this problem as an active feedback framework and derived several practical algorithms based on the diversity of the feedback documents. Their algorithms take into account of the document diversity by clustering retrieved documents or choosing documents with a certain ranking gap. In our paper, we proposed a new active feedback approach which comprehensively considers relevance, diversity and density of the feedback documents. We call this new active feedback algorithm **Active-RDD** (denoting Active Learning to achieve **Relevance**,**Diversity** and **Density**).

Active feedback is essentially an application of active learning in ad hoc information retrieval. Active learning has been extensively studied in supervised learning and other related context. Cohn et al. [4] proposed one of the first statistical analysis of active learning, demonstrating how to construct queries that maximize the error reduction by minimizing learners' variance. They developed their method for two simple regression problems in which this question can be answered in closed form. Both the Query by Committee (QBC) algorithm [5] and Tong's version space method [6] are based on choosing a sample which is close to classification boundary. Both of their methods have been applied to text classification problems. To avoid choosing outliers, McCallum and Nigam [7] modify the QBC method to use the unlabeled pool for explicitly estimating document density. Batch mode active learning, which selects a batch of unlabeled examples simultaneously, is an efficient way to accelerate the learning speed. In [8], Brinker presented a new approach that is especially designed to construct batches by incorporating a diversity measure. Besides the above application area, supervised learning, active learning has also been recently applied to adaptive information filtering [9].

One major drawback of the above methods is their computational complexity, which prevents us from using them directly in the information retrieval task. This paper explores how to overcome this problem by designing an efficient active learning algorithm (Active-RDD) for relevance feedback. Because most of the well motivated active learning approaches choose data samples by implicitly or explicitly considering the uncertainty, density or diversity of data samples, we designed the new algorithm to explicitly capture these important factors by integrating document relevancy, document density measure and document diversity measure. We apply the proposed algorithm to the language modeling retrieval framework and evaluate the effectiveness of the proposed technique on two benchmark data sets. The experimental results demonstrate the statistical validated performance improvement of our algorithm over existing algorithms.

The remainder of this paper is organized as following. In section 2, we first analyze the important elements that influence retrieval performance and derive an efficient active learning algorithm for document selection based on these elements. In section 3, we discuss the experimental setting and the experimental results. In Section 4, we conclude with a description of our current research, and present several future research directions for further work.

2 Active Learning Algorithm

2.1 Algorithm Intuition

The goal of active relevance feedback is to improve retrieval performance by actively selecting feedback documents for user evaluation. Here we will first illustrate the intuition underlying our new approach.

Relevant documents directly reflects a user's search interest, and the current relevance feedback algorithms based on language modeling only rely on the information contained in relevant feedback documents. So choosing relevant documents for evaluation will effectively direct the second round search results to the user's intent. Initially, when a query is input into a retrieval engine, we do not know the true relevance of documents until we get feedback from the user. The only criteria to judge the relevance of a document during an initial pass is the relevance score given by retrieval engine. The relevance score of a document is calculated based on the similarity between the initial query and the document. Considering the above two facts, we will choose documents with high relevance scores. The traditional relevance feedback method Top K selects the top k ranked documents for feedback. Although the Top K algorithm is in line with our hypothesis, which is that relevant documents are good for learning, it is not the best strategy from a learning perspective. For instance, if there are two identical documents among the top ranked documents, the improvement of second round retrieval performance achieved by choosing both documents is equivalent to the improvement achieved by choosing any one of them. In the next section, we will analyze another important factor on choosing feedback documents to avoid this redundancy problem in the previous example.

The Top K approach does not take into account of the redundancy between selected feedback documents: this redundancy results from very similar (and near duplicated) documents. Thus, in our active learning approach, we need to capture diversity of feedback document set in the algorithm. The Gapped Top K algorithm [3] increases the diversity of feedback documents by selecting the top K documents with a ranking gap G in between any two documents. Another heuristic method to increase diversity is the Cluster Centroid algorithm [3], which groups retrieved documents into K clusters and chooses one representative document from each cluster. Our Active-RDD algorithm, which is different from the above two methods, maximizes the diversity of feedback document set by explicitly maximizing the distance between new document and selected documents.

local consideration

K聚类

考虑训练集

If the selection criterion only takes into account the relevance score and diversity of the batch document set, it loses the benefit of the implicit modeling of the data distribution. For instance, such selection criteria may select documents that lie in unimportant, sparsely populated regions. Labeling documents in high density regions or in low density regions gives the query feedback algorithm different amounts of information. To avoid choosing outliers, we aim to select documents in high density regions. Choosing relevant documents in high probability density regions will retrieve more relevant documents in the subsequent round, which leads to a better retrieval performance.

Finally, in order to combine the above three factors, we build a linear combination of all the measures and proceed in the following way to construct a new feedback document set. To reduce the computation, we select K feedback document from the top L ranked documents. For instance, the reasonable sizes of L and K could be 100 and 6 respectively. Let I denote the set of unlabeled documents that have not yet been selected for evaluation, we incrementally construct a new feedback document set S. The selection scheme can be described as follows: 筛

```
$S = 0$
repeat
    $d_i = \arg\max_{d_i \in I \notin S} [(\alpha)\text{relevance}(d_i) + (\beta)\text{density}(d_i) + (1-\alpha-\beta)\text{diversity}(d_i, S)]$   (1)
    $S = S \cup d_i$
Until $size(S) = K$
```

where relevance(d_i) is the relevance score of document d_i, density(d_i) is the density performance measure around document d_i, and distance(d_i, S) is the distance between document d_i and the existing feedback document set S . $\alpha \in [0, 1]$, $\beta \in [0, 1]$ are weighting parameters. Setting $\alpha = 1$ restores the Top K approach; if $\beta = 1$, the algorithm selects feedback document only based on its density performance measure; whereas if $\alpha = 0$ and $\beta = 0$, the algorithm focuses exclusively on maximizing the diversity of selected document set. In the following sections, we will explain how we calculate the above three factors in detail.

2.2 Relevance Measure

Language modeling approaches to information retrieval have received recognition for being both theoretically well founded, and showing excellent retrieval performance and effective implementation in practice. In this paper, we apply language modeling approach using KL divergence measure for our basic retrieval model. Suppose that a query q is generated by a generative model $p(q|\theta_Q)$ with θ_Q denoting the parameters of the query unigram language model. Similarly, we assume that a document d is generated by a generative model $p(d|\theta_D)$ with θ_D denoting the parameters of the document unigram language model. The query unigram language model and document unigram language model are smoothed multinomial models in language modeling. If $\widehat{\theta}_Q$ and $\widehat{\theta}_D$ are the estimated query language model and document language model respectively, then the relevance score of document d with respect to query q can be calculated by negative KL-divergence[10]. KL-divergence is calculated by the formula below:

$$KL(\widehat{\theta}_Q \| \widehat{\theta}_D) = \sum_w p(w|\widehat{\theta}_Q) \log \frac{p(w|\widehat{\theta}_Q)}{p(w|\widehat{\theta}_D)} \qquad (2)$$

Where $p(w|\widehat{\theta}_Q)$ is the probability of generating word w by query language model $\widehat{\theta}_Q$; $p(w|\widehat{\theta}_D)$ is the probability of generating word w by document language model $\widehat{\theta}_D$.

The retrieval engine ranks all the documents according to their negative KL-divergence scores. In the Active-RDD algorithm, we use the negative KL-divergence measure, which is given by first round search, as relevance score.

2.3 Document Density Measure

Document density is one of the important factors in the defined active selection scheme. Owing to the large scale of the document collection, estimating document probability density in the whole collection is computationally unachievable. To reduce the computation, we only measure the density performance of the top L documents in the initial retrieval results.

We approximate the density in a region around a particular document by measuring the average distance from that document to all the other documents. Distance between individual documents is measured by J-Divergence[11]. KL divergence is a non symmetric measure between two probability mass functions, while J-Divergence obtains the symmetry by adding two KL divergences as described in (2). The formula of J-Divergence is as follows:

可以考虑对于biaffine也采用同样的策略，让其对称

$$J(d_i||d_j) = KL(d_i||d_j) + KL(d_j||d_i) \tag{3}$$

Consequently, the average J divergence between a document d_i and all other documents measures the degree of overlap between d_i and all other documents. In other words, large average J divergence indicates that the document is in low document density region. Thus we use negative average J divergence (4) to approximate document density performance measure, which reflects the closeness of this document to the other documents. The reason we use this measure is to normalize the value of density performance measure to be on the same scale of the relevance score.

代表性是平均距离

不确定性和代表性的范围要一样

$$\text{density}(d_i) = \frac{-1}{|D|} \sum_{d_h \in D} J(d_i||d_h) \tag{4}$$

2.4 Diversity Measure

The metric we use to measure the distance between a document and a document set is the minimum distance between the document and any document in the set. This method corresponds to the single linkage method in hierarchical clustering literature. The single linkage method has the advantage of efficient time complexity, and it also ensures that the new document is different from all the selected documents.

局部考量就相当于单链接聚类？

To normalize all components in the overall metric to be of comparable values, we use J divergence to measure the distance between candidate document and selected documents. To maximize the combined score of relevance score, density performance measure and diversity measure, which is shown in (1), we employ the following incremental strategy: Given a set of unlabeled documents, we start with document d_1 which has the highest combined score of relevance score and density performance measure; then we add a new document d_2 to our set $S = d_1 \cup d_2$, which maximize the combined score of relevance score, density performance measure and diversity measure. We continue by adding new documents until the size of the selected documents reaches the predefined size.

The individual influence of each factor can be adjusted by the weighting parameters α and β. The combined strategy can be implemented very efficiently. Recalculating the distance between an unselected document and every single document already added in the feedback document set to evaluate the maximum distance between the unselected document and the document set results in quadratic computational time depending on the feedback document size. We cache the maximum distance of all the unselected documents from selected document set and update the score only if the distance between the newly added document and the unselected document is larger than the stored maximum. We only need to compute distance once for every unselected document instead of already selected documents number. If we are choosing K documents from top L retrieved documents, the computation complexity in this part is reduced from $O(K^2L)$ to $O(KL)$. The complete pseudo code of an efficient implementation of the algorithm is given in Table 1.

关于时间复杂度的讨论

The Maximal Marginal Relevance ranking algorithm [12] (MMR) is a greedy algorithm for ranking documents based on relevance ranking score and at the same time avoiding redundancy. Our Active-RDD algorithm extends the MMR algorithm by adding an extra term, which reflects the document density. In [3], Shen and Zhai proposed the MMR algorithm to solve the active feedback problem, but they have not implemented that algorithm.

也太坑了吧

2.5 Query Updating Algorithm

Based on user's relevance judgment on feedback document, we use the divergence minimization model [13] to update query. The divergence minimization model minimizes the divergence between the query model and the relevant feedback documents. Let $R = d_1, \ldots, d_n$ be the set of relevant feedback documents. We define the empirical KL-divergence between the feedback query model θ_F and the relevant feedback documents $R = d_1, \ldots, d_n$ as the average divergence between the query model and relevant feedback document model.

$$D_e(\theta_F, R) = \frac{1}{|R|} \sum_{i=1}^{n} D(\theta_F \| \theta_i) \tag{5}$$

We subtract the negative divergence between the query language model and collection model to remove the background information. Considering all the above conditions, we derive the following empirical divergence function of a feedback query model:

$$\theta_F = \arg\min_{\theta_F} \frac{1}{|R|} \left\{ \sum_{i=1}^{n} D(\theta_F \| \theta_i) - \lambda D(\theta_F \| p(.|C)) \right\} \tag{6}$$

Table 1. Active-RDD Algorithm

```
input:
$\alpha$                                   (relevance coefficient)
$\beta$                                    (density coefficient)
$K$                                        (size of feedback document set for evaluation)
$L$                                        (size of document set from which we choose $K$ documents)
$D = (d_0, \dots d_{L-1})$                 (permutation of $0, \dots, L-1$)
$R = (r_0, \dots r_{L-1})$                 (relevance score of each document)
output:
$D = (d_0, \cdots d_{L-1})$                (permutation of $0, \dots, L-1$)
    relevance = $array[L]$
    maxDis = $array[L]$
    for $j = 0$ to $L-1$ do
        relevance($j$) = $R(j)$
        Calculate document density performance using (4)
        maxDis($j$) = 0
    end for
    for $k = 0$ to $K-1$ do
        maxIndex = $k$
        maxValue = 0
        for all $j = k$ to $L$ do
            value= ($\alpha$) relevance($j$) + ($\beta$)density($j$) + $(1 - \alpha - \beta)$maxDis($j$)
            if value > maxValue then
                maxValue = value
                maxIndex = j
            end if
        end for
        swap ($d_{\text{maxIndex}}, d_k$)
        for all $j = k+1$ to $L$ do
            distance = $J(d_j||d_k)$
            if distance > maxDis(j) then
                maxDis($j$) = distance
            end if
        end for
    end for
```

Here $p(.|C)$ is the collection language model and $\lambda \in [0, 1)$ is the weighting parameter. Taking the first derivative of (6) with respective to $p(w|\theta_F)$, we will get the simple closed form solution.

$$p(w|\theta_F) \propto \exp(\frac{1}{1-\lambda}\frac{1}{|R|}\sum_{i=1}^{n}\log p(w|\theta_i) - \frac{\lambda}{1-\lambda}\log p(w|c)) \tag{7}$$

To exploit θ_F in our KL-divergence retrieval model, we interpolate it with the original query model θ_Q to obtain updated model θ'_Q ,

$$\theta'_Q = (1-\mu)\theta_Q + \mu\theta_F \tag{8}$$

and then use the updated query θ'_Q to score document d_i by negative KL-divergence.

3 Experiment Methodology and Experimental Results

To evaluate our Active-RDD algorithm described in previous sections, we use two different TREC data sets. The first one is TREC HARD 2005 Track, which contains the full AQUAINT collection; the second one is TREC HARD 2003 Track, which use part of AQUAINT data plus two additional datasets (Congressional Record (CR) and Federal Register (FR)). We do not have the additional data set in TREC HARD 2003 Track. Our results are comparable to other published TREC HARD 2003 results, although the data is a little different. For both tracks, we use all the 50 topics which have relevance judgments. We use only the titles of the topic description, because they are closer to the actual queries used in real applications.

We employ the Lemur Toolkit[14] as our retrieval system and KL-Divergence language retrieval model as our baseline retrieval model. We compare the Active-RDD algorithm with the existing active feedback algorithms such as Top K, Gapped Top K and Cluster Centroid. For all the algorithms, we select $(K) = 6$ feedback documents from top $(L) = 100$ documents. All the parameters in the query updating model are fixed at the default values in The Lemur Toolkit[14].

To measure the performance of an active relevance feedback algorithm, we use two standard ad hoc retrieval measures: (1) Mean Average Precision (MAP), which is calculated as the average of the precision after each relevant document is retrieved, reflects the overall retrieval accuracy. (2) Precision at 10 documents (Pr@10): this measure does not average well and only gives us the precision for the first 10 documents. It reflects the utility perceived by a user who may only read up to top 10 documents.

In the following sections, we use cross-validation for Active-RDD algorithm and Gapped Top K algorithm, and then statistically compare the Active-RDD algorithm with existing algorithms.

3.1 Cross Validation

Coefficients α and β play an important role on selecting the feedback documents. How to select these coefficients significantly impacts the overall algorithm performance. In order to have a fair comparison, we pursue 5-fold cross-validation on the Active-RDD algorithm and Gapped Top K algorithm, and compare their cross-validation performance (CVP) with Cluster Centroid and Top K algorithm performance,(these algorithms are consequently parameter free in this setting).

We separate 50 queries into 5 parts, where each part contains 10 queries. For the kth set of queries, we train parameters to optimize the retrieval performance for the other 4 sets of queries, and use this set of the parameters to test on kth set of queries to obtain the retrieval performance measure for kth part.

We do this for $k = 1, 2, 3, 4, 5$ and the cross-validation performance is the average performance on the 5 test query sets. The cross-validation experimental results are shown in Table 2.

From Table 2, we conclude that the cross-validation performance of our Active-RDD algorithm is better than the Gapped Top K algorithm. Furthermore, we will compare these cross-validation performances with the Cluster Centroid algorithm and Top K algorithm.

Table 2. Cross-validation comparison of Active-RDD and Gapped Top K approaches. CVP indicates cross-validation performance, which is the average value of the MAP and Pr@10 on test data.

	Active-RDD				Gapped Top K			
HARD 2003	MAP	MAP	Pr@10	Pr@10	MAP	MAP	Pr@10	Pr@10
	Train	Test	Train	Test	Train	Test	Train	Test
Folder 1	0.3855	0.3566	0.5925	0.6700	0.3676	0.3295	0.5450	0.6400
Folder 2	0.3954	0.3169	0.6325	0.5100	0.3792	0.2831	0.5950	0.4400
Folder 3	0.3966	0.3119	0.6225	0.5300	0.3747	0.3013	0.5925	0.4500
Folder 4	0.3793	0.3812	0.6275	0.5500	0.3594	0.3189	0.5750	0.5100
Folder 5	0.3416	0.5319	0.5650	0.7800	0.3175	0.5299	0.5275	0.7100
CVP		0.3797		0.6080		0.3525		0.55
HARD 2005	MAP	MAP	Pr@10	Pr@10	MAP	MAP	Pr@10	Pr@10
	Train	Test	Train	Test	Train	Test	Train	Test
Folder 1	0.2675	0.2356	0.5575	0.5400	0.2496	0.2634	0.5450	0.6400
Folder 2	0.2583	0.2722	0.5550	0.5700	0.2309	0.2821	0.5525	0.6100
Folder 3	0.2489	0.3097	0.5325	0.6400	0.2508	0.2584	0.5600	0.5800
Folder 4	0.2673	0.2362	0.5700	0.4900	0.2594	0.2238	0.5875	0.4700
Folder 5	0.2634	0.2519	0.5600	0.5300	0.2569	0.2339	0.5750	0.5200
CVP		0.2611		0.5540		0.2523		0.5640

3.2 Comparison of Different Active Learning Algorithms

To evaluate the effectiveness of different document selecting approaches, we compare the performance of the non-feedback approach baseline with Top K, Gapped Top K , Cluster Centroid and our Active-RDD algorithm, all of which are feedback based algorithms. The performance of the Active-RDD and the Gapped Top K algorithm are the cross-validation performance in the previous section.

From Table 3, we can see that all these feedback algorithms perform better than the baseline non-feedback retrieval. All the results show that the underlying relevance feedback mechanism is very effective. From the results, our active learning algorithm Active-RDD outperforms Top K algorithm significantly, and it also performs better than other active feedback approaches at the statistical significance level 10% in most cases.

Table 3. Average performance of different active learning approaches. The best performance is shown is bold. We compare our Active-RDD algorithm with the Top K algorithm, the Gapped Top K algorithm and the Cluster Centroid algorithm, and percentage improvements over these three existing algorithms are shown in column 7,8,9 respectively. A double star(**) and a single star(*) indicate that the performance of our active learning algorithm is significantly better than the existing method used in the corresponding column (Top K, Gapped Top K or Cluster Centroid) according to Wilcoxon signed rank test at the level of 0.05 and 0.1 respectively.

Method		Baseline	Top K	Gap K	Cluster	RDD	Improv. over Top K	Improv. over Gap K	Improv. over Cluster
HARD 2003	MAP	0.3150	0.3508**	0.3525**	0.3771	**0.3797**	8.07%	7.72%	0.69%
	pr@10	0.5000	0.5380**	0.5500**	0.5760**	**0.6080**	13.01%	10.55%	5.56%
HARD 2005	MAP	0.1919	0.2367**	0.2523	0.2369*	**0.2611**	10.31%	3.49%	10.22%
	pr@10	0.4340	0.4800**	**0.5640**	0.5420**	0.5540	15.42%	−1.77%	2.21%

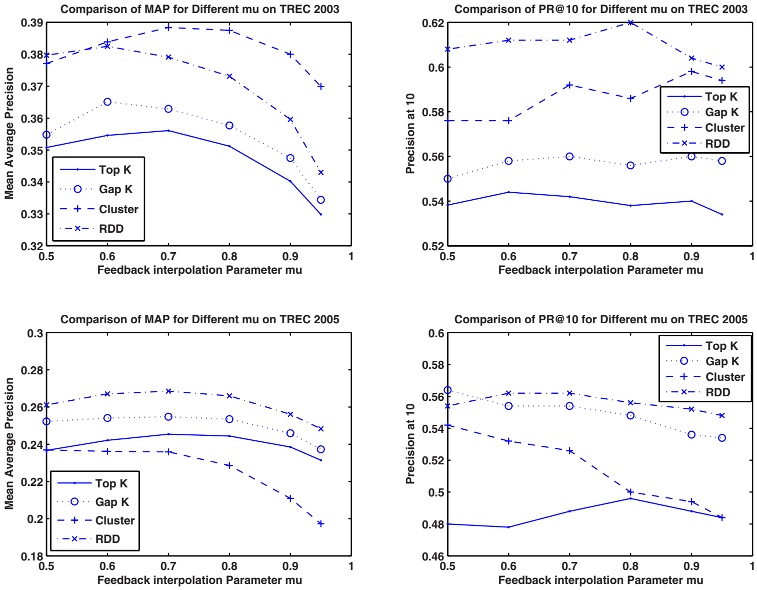

Fig. 1. Sensitivity of average performance of different active learning algorithm on μ

3.3 Performance Sensitivity of Feedback Interpolation Parameter μ

Owing to the nature of explicit feedback, the relevant feedback documents judged by the user are more reliable. This intuition leads to adding more weight to the feedback interpolation parameter μ in (8). In the previous experiments, we set $\mu = 0.5$ as the Lemur Toolkit[14] default setting. We did another set of experiments by increasing μ, and the results are shown in Fig. 1. The results indicate that setting $\mu = 0.7$ gives the Active-RDD algorithm best performance (with performance improvement of $1-2\%$). The curves are fairly flat and indicate relative insensitivity around the optimal value of feedback parameters, which is a desirable pattern.

4 Conclusions

This paper explores the problem of how to select a good set of documents to ask user for relevance feedback. This paper presents a new efficient active learning algorithm, which dynamically selects a set of documents for relevance feedback based on the documents' *relevancy*, *density* and *diversity*. We evaluate the algorithm on TREC2005 HARD dataset and TREC2003 HARD dataset. The experimental results show that our algorithm significantly outperforms the existing Top K, Gapped Top K and Cluster Centroid algorithms.

There are several interesting research directions that may further improve relevance feedback under the active learning framework: first, making full use of users' feedback by learning from non-relevant documents; second, learning different active learning parameters for different queries; and third, combining implicit feedback with active learning.

Acknowledgments

We would like to acknowledge support by Cisco, University of California's MICRO Program, CITRIS, and UARC. We also appreciate discussions with associated colleagues.

References

1. Harman, D.: Relevance feedback revisited. In: Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (1992) 1–10
2. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. Journal of the American Society for Information Science **41**(4) (1990) 133–168
3. Shen, X., Zhai, C.: Active feedback in ad hoc information retrieval. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. (2005) 55–66
4. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. In: Advances in Neural Information Processing Systems. Volume 7., The MIT Press (1995) 705–712
5. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning **28**(2-3) (1997) 133–168
6. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In: Proceedings of 17th International Conference on Machine Learning. (2000) 999–1006

7. McCallum, A., Nigam, K.: Employing EM and pool-based active learning for text classification. In: Proceedings of the Fifteenth International Conference on Machine Learning. (1998) 350 – 358
8. Brinker, K.: Incorporating diversity in active learning with support vector machines. In: Proceedings of the Twentieth International Conference on Machine Learning . (2003) 59–66
9. Zhang, Y., Xu, W., Callan, J.: Exploration and exploitation in adaptive filtering based on bayesian active learning. In: Proceedings of 20th International Conf. on Machine Learning. (2003) 896–903
10. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In: Research and Development in Information Retrieval. (2001) 111–119
11. Lin, J.: Divergence measures based on the shannon entropy. IEEE Transactions on Information Theory (1) (1991) 145–151
12. Carbonell, J.G., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (1998) 335–336
13. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of the Tenth ACM International Conference on Information and Knowledge Management. (2001) 403–410
14. (The lemur toolkit) http://www.lemurproject.org.