基于 LDA 主题模型的分布式信息检索集合选择方法

何旭峰[1]，陈岭[1]，陈根才[1]，钱坤[1]，吴 勇[2]，王敬昌[2]

[1]（浙江大学计算机科学与技术学院， 杭州 中国 310027）

[2]（浙江鸿程计算机系统有限公司， 杭州 中国 310009）

摘 要：针对分布式信息检索时不同集合对最终检索结果贡献度有差异的现象，提出一种基于 LDA 主题模型的集合选择方法。该方法首先使用基于查询的采样方法获取各集合描述信息；其次，通过建立 LDA 主题模型计算查询与文档的主题相关度；接着，用基于关键词相关度和基于主题相关度结合的方法估计查询与样本集中文档的综合相关度；最后，通过样本集文档所属的集合信息，估计查询与各集合的相关度，进而选择相关度最高的 M 个集合进行检索。实验部分采用 R_m、$P@n$ 和 MAP 作为评价指标对集合选择方法的性能进行了验证。实验结果表明本文提出方法能更准确的定位到包含相关文档多的集合，提高了检索结果的召回率和准确率。

关键词：集合选择；分布式信息检索；LDA

中图法分类号：TP 311

A LDA topic model based collection selection method for distributed information retrieval

HE Xu-feng[1], CHEN Ling[1], CHEN Gen-cai[1],QIAN Kun[1],WU Yong[2], WANG Jing-chang[2]

[1](College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China)

[2](ZheJiang Hongcheng Computer System Co.,Ltd., Hangzhou 310009, China)

Abstract: Considering that different collections have different contributions to the final search results, a LDA topic model based collection selection method was proposed for distributed information retrieval. Firstly, the method acquired information about the representation of each collection by query-based sampling; secondly, a method using the LDA topic model was proposed to estimate the relevance between the query and a document; then, a term-based and topic-based mixed method was used to estimate the relevance between the query and the documents sampled; Finally, the relevance between the query and collections were estimated with the information of the collections that the sampled documents belong to, and M collections with the highest relevance were selected for retrieving. R_m, $P@n$ and MAP were used to evaluate the effectiveness of the collection selection method. Experiment results demonstrated that the proposed method was better at selecting collections with more relevant documents, and can improve the accuracy and recall of search results.

Key words: Collection Selection; Distributed Information Retrieval; LDA

1 引言

随着网络信息的膨胀，传统的集中式信息检索在面对海量数据时往往不堪重负。分布式信息检索系统[1]（distributed information retrieval，DIR）将大范围分布的异构数据联合起来，形成一个逻辑整体，为用户提供强大的信息检索能力，被用来解决海量数据的存储检索问题。DIR 系统通常由一个代理服务器和很多检索服务器组成。代理服务器提供统一的查询接口，负责将查询请求分发给各检索服务器。

将查询请求发送给所有信息集检索将导致大量的网络带宽消耗和较低的系统响应速度。集合

资助项目：国家“核高基”重大科技专项课题（2010ZX01042-002-003）；国家自然科学基金（60703040，61332017）；浙江省重大科技专项（2011C13042，2013C01046）；中国工程科技知识中心资助项目（CKCEST-2014-1-5）

通信作者：陈岭，E-mail：lingchen@zju.edu.cn

选择，就是对每个查询请求，在代理服务器上决策出选择哪些信息集去检索，使得在较少的信息集合中检索，得到尽可能多的相关文档。

近十年来在分布式信息检索集合选择领域出现了许多研究，这些研究大抵可分为三类：

1）将集合中包含的文档看成一个逻辑整体，一个集合即是一个超大“文档”（big document），将计算查询与集合的相关度就转化为计算查询与文档相关度，按相关度的高低对各集合排序。

CORI[2]方法采用 INQUERY 推理网络计算集合相关度，关键词的权重用 TF-IDF 的方法计算。然后利用各集合与查询包含的词和词的权重，计算各集合与查询的相关度。Xu 和 Croft[3]建立语言模型并用集合和查询词的 KL 距离（Kullback-Leibler Divergence）计算集合相关度，Si[4]等人基于语言模型以单个集合为语料库，通过各个语料库生成查询词的概率计算集合相关度。Yuwono 等人提出了 CVV 方法[5]，该方法引入了线索有效性（Cue Validity）的概念来描述词 t 对属于集合 C_i 中的文档和属于其它集合的文档的区分度。以上方法将各个集合中的所有文档看成一个逻辑上的统一整体，从而将分布式的集合选择问题转化为传统的查询词对文档的检索问题，然而由于虚拟文档与真实的文档之间实际上存在着很大的不同，如文档长度，主题等，以上方法倾向于选择小的，主题较少的集合，而不是大的，主题复杂，包含相关文档更多的集合，不适用于集合大小不均匀的环境。

2）估算集合中包含的与查询相关的文档数目，以相关文档数目的多少对各集合排序，从而选择排名较高的集合检索。

ReDDE[6]，CRCS[7]，通过基于查询采样方法对各集合采样，构建中心样本，通过查询词和被采样文档的信息，估计各个集合和文档的相关度。SUSHI[8]方法在 ReDDE 方法以及 CRCS 方法的基础上，通过曲线拟合的方法，用中心样本集检索结果中各文档的得分，估计原集合中各文档的得分，从而得到各集合的得分。以上方法通过估计各个集合包含的与查询相关的文档数，按包含文档数目从高到低对集合排序，能够有效定位到包含相关文档数目更多的集合然而，对于返回 top N 个结果的检索请求，与查询最相关的文档未必包含在相关文档数目最多的集合，从而不会被检索到。

3）计算集合与查询的相关度得分，按得分的高低对集合排序。

SHiRE[9]将查询检索中心样本的结果构造成树结构，通过三种不同的遍历该树的次序的方法 Lex-S、Rank-S 和 Conn-S，分别得到三种不同的确定样本集检索结果影响的指数衰减数的方法，并以此指数衰减数确定文档对应集合的得分。

Cetintas 等人提出基于历史查询的集合选择方法[10]，通过历史查询结果指导新查询，新查询利用历史相似查询的结果进行集合选择。刘颖等人提出了基于历史点击数据的集合选择方法[11]，通过历史查询的用户点击结果，以及新查询与历史查询的相似度，估计新查询与各集合的相关度。然而由于词语普遍存在二义性且查询比较简短，查询并不总能准确的表达出用户的信息需求，导致一些相关度得分高的文档和查询并不相关。

Wauer 等人[12]提出了基于本体论的方法，该方法首先选取一组特定的主题，然后将查询与集合分别表示成关于这一组主题的向量，利用向量余弦计算相关度。但是该方法只能在特定的、主题集中的数据集下选取一组特定的主题，很难推广到其它数据集。

针对以上问题，本文提出基于 LDA[13]主题模型的集合选择方法 LBCS，主要工作如下：

1）在计算查询与样本集文档的相关度中，提出了基于关键词相关度和基于主题相关度结合的方法来估计查询与样本集中各文档的综合相关度的计算方法。

2）在计算查询与文档的主题相关度时，充分考虑二者潜藏的语义关系，提出了基于 LDA 主题模型的计算方法。

3）在得到查询与样本集中文档的综合相关度后，考虑到各文档所属的原集合信息，提出了计算查询与集合相关度的方法。

4）设计实验对比了本文提出的 LBCS 集合选择方法和 ReDDE、CRCS 方法在测试集上的检索召回率和准确率，通过实验数据说明了方法的性能。

1 LBCS 集合选择方法

1.1 分布式信息检索架构

分布式信息检索系统，如图 1 所示，包含一组客户端，一个检索代理服务器和一组检索服务器，其中检索服务器中存放着各自的信息集合，各检索服务器独立的检索自己的信息集合。

一个具体的查询过程可以描述如下：用户通过查询客户端，将查询请求发送到检索代理服务器；检索代理服务器通过集合选择方法选择出与该查询相关的部分集合，并将检索请求分别发送至选择的检索服务器；各检索服务器单独的在自己的信息集合中检索，然后把检索结果返回给代理服务器；检索代理服务器收到各检索服务器的检索结果后，通过合并策略对得到的检索结果合并，并进行全局的排序；最后检索代理服务器将排序后的检索结果返回到查询客户端，呈现给用户。

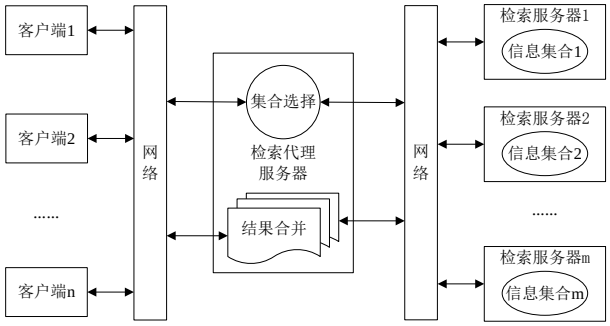

图 1 分布式检索体系结构图

分布式信息检索包含三个主要任务：

1）集合描述信息获取，分布式检索代理服务器获取个信息集合的描述信息，描述信息包含该集合是否可被检索，集合中包含的文档内容（文档频率，词频等）等信息。

2）集合选择，由于并非所有集合包含用户所需的信息，将查询发给所有的检索服务器浪费计算及通信资源，因此，需要选择和查询最相关的信息子集进行检索。

3）结果合并，将不同检索服务器返回的检索结果需经过检索代理按一定规则合并后返回给用户。

本文主要关注第二个任务，集合选择问题。

1.2 LDA 主题模型

LDA 主题模型，是一种完全的概率生成模型，目的是要以一种非监督的机器学习方法，由信息集各文档的词与词频的静态信息学习出信息集中文档与主题、主题与词的概率模型，然后按照学习出的概率模型可以有效的识别语料库或大规模文档集中隐藏的主题信息。LDA 通过基于词袋的方法，将文档的文本信息转为成了形式化的数学统计信息。LDA 生成文档集中的各个文档的过程可以描述如下：

1）首先从文档集中任意选取一篇未读文档 di；

2）基于狄利克雷参数 α，从“文档-主题”的概率分布 θ 中选取一个与文档 d_i 相关的主题 z_k；

3）对于选取到的主题 z_k，基于参数 β，从“主题-词语”的概率分布 φ 中选取一个与该主题相关的词 t_j；

4）重复 2）、3）过程直到生成了文档 d_i 中的所有的词，将文档 d_i 标记为已读；

5）重复 1）、2）、3）、4）过程直到遍历完文档集中所有文档。

文档在主题空间服从多项式分布 θ，主题和文档集中的所有词服从多项式分布 φ，其中 θ 和 φ 的先验分布是狄利克雷分布，分别带有参数 α 和 β。

LDA 主要有模型作者提出的变分-EM 算法[13]，还有现在常用的 Gibbs 抽样法[14]来推断其参数，其结果包括文档-主题分布 $P(z|d, \theta)$，主题-词分布 $P(w|z, \varphi)$。本文使用吉布斯抽样作为 LDA 参数的推理方法，利用 LDA 推断出的文档-主题分布与主题-词分布，计算查询词和文档的主题相关度。Wei[15]指出 LDA 主题模型能有效应用到信息检索中，提高信息检索方法的准确率。

1.3 LBCS 方法流程

LBCS 的总体流程图如图 2 所示，分为在线和离线两部分，离线部分包括：1）代理服务器使用基于查询的采样方法[16]对各集合采样，对于每个查询，各个集合返回前三个检索结果，在代理服务器上构建样本集；2）代理服务器对样本集预处理，构建倒排索引，同时对样本集建立 LDA 主题模型，推导出主题-词分布 φ，以及文档-主题分布 θ。在线部分包括：1）查询检索倒排索引，计算查询与各文档关键词相关度（计算方法见 1.4.1 节）；2）通过历史查询得到新查询的扩展查询，同时利用扩展查询和 LDA 推断出的分布计算查询与各文档主题相关度（计算方法见 1.4.2 节）；3）得到查询与样本集中各文档的综合相关度（计算方法见 1.4 节），并按相关度高低对各文档排序，然后以此计算查询与各个集合的相关度（计算方法见 1.5 节），并按相关度高低对各集合排序；4）选择排序结果最靠前的 M 个集合，将检索请求发送到这些集合。

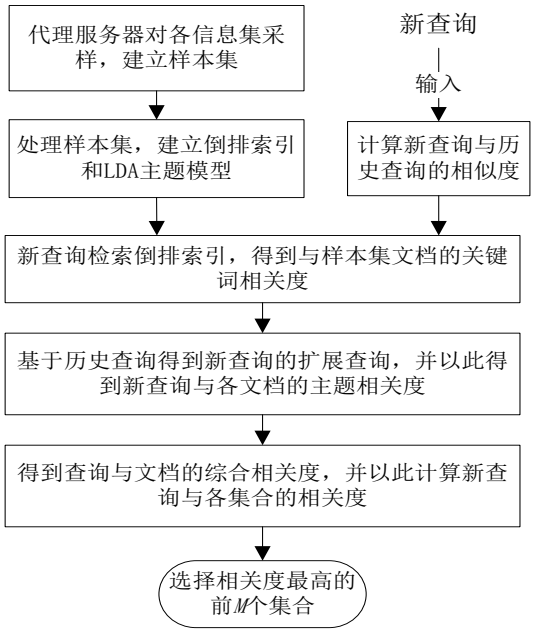

图 2 LBCS 方法流程图

在实际的系统中，各检索服务器的信息集合的内容不会一成不变，因而离线预处理部分需要定期的触发执行，从而不断的更新样本集合，为集合选择提供依据。而基于查询的采样方法和基于 LDA 的主题建模方法需要消耗不短的时间，为了保证在进行数据预处理时，检索系统能正常提供服务，需要离线的执行数据预处理模块。即假

设当前存在样本集合 Sample_1，集合选择利用样本集合 Sample_1 的数据提供服务，当触发数据预处理模块时，集合选择继续使用样本集合 Sample_1 的数据，直到数据预处理完成得到建模后的样本集合 Sample_2，集合选择再切换到使用样本集合 Sample_2 的数据，然后删除原样本集合 Sample_1。

1.4 查询与样本集文档相关度

定义 1. 扩展查询。查询 $q=\{t_1, t_2, ..., t_m\}$，另一组查询为$\{p_1, p_2, ..., p_n\}$，其中每个查询 p_i 都可以表示为一组词的形式 $p_i=\{t_{pi\_1}, t_{pi\_2}, ..., t_{pi\_m_i}\}$，则查询 q 基于这组查询$\{p_1, p_2, ..., p_n\}$的扩展查询 $q' = \{t_1, t_2, ..., t_m, t_{p1\_1}, t_{p1\_2}, ..., t_{p1\_m_1}, ..., t_{pn\_1}, t_{pn\_2}, ..., t_{pn\_m_n}\}$，扩展查询 q'中的每个词的影响度为该词所属的原查询与查询 q 的相似度，即对于词 $t \in q'$且 $t \in p_i$，词 t 在扩展查询 q'中的影响度 eff_t = sim($q|p_i$)，其中 sim($q|p_i$)为计算查询 p_i 与查询 q 相似度的函数。

定义 2. 词与文档主题相关度。一个词 t 与一组主题$\{z_1, z_2, \ldots, z_k\}$相关的概率为$\{P_{t_1}, P_{t_2}, \ldots, P_{t_k}\}$，文档 d 与这组主题相关的概率为$\{P_{d_1}, P_{d_2}, \ldots, P_{d_k}\}$，那么词 t 与文档 d 的主题相关度，即对于主题$\{z_1, z_2, \ldots, z_k\}$，$t$ 和 d 相关的概率 $P(t, d) = P_{t_1} \times P_{d_1} + P_{t_2} \times P_{d_2} + \ldots + P_{t_k} \times P_{d_k}$。

定义 3. 查询与文档主题相关度。一个查询 q 由一组词$\{t_1, t_2, \ldots, t_j\}$构成，这组词对查询 q 的影响度分别为$\{w_1, w_2, ..., w_j\}$，查询与文档的主题相关度即为查询中的各个词与文档的主题相关度以及该词在查询中的影响度之积的累加和，即 $P(q, d) = P(t_1, d) \times eff_1 + P(t_2, d) \times eff_2 + \ldots + P(t_j, d) \times eff_j$。

LDA 提供了一种新的基于主题的文档建模方式，然而仅仅使用 LDA 本身用于检索过于粗糙而影响检索结果，使用 LDA 模型与传统检索方法结合有利于提高检索准确率。

文档与查询的得分由两部分线性组成，一部分是基于 LDA 主题模型的文档和查询的主题相关度，另一部分是基于词的文档与主题的关键词相关度。样本集当中的各文档与查询 q 的综合相关度：

$$Score(d_i,q) = \lambda \times Score_{lda}(d_i,q) + (1-\lambda) \times Score_{keyword}(d_i,q) \quad (1)$$

其中 λ 取值在[0, 1]。各文档按照相关度排序，排序的结果用于估计集合和查询的相关度。

1.4.1 查询与样本集文档关键词相关度

将查询看成由查询词构成的短文本，查询 q 和文档可表示为向量($<t_1, W_{t1}>, <t_2, W_{t2}>, \ldots, <t_n, W_{tn}>$)，式中 t_i 为查询或文档的第 i 个词，W_{ti} 为 t_i 的权重。查询与文档的关键词相似度采用空间向量模型的余弦相似度计算，权重使用 *tf-idf* 的方法计算，并将结果归一化：

$$rel(q \mid d_i) = \frac{\sum_i W_{ti,q} \times W_{ti,d}}{\sqrt{\sum_i W_{ti,q}^2} \times \sqrt{\sum_i W_{ti,d}^2}} \quad (2)$$

$$W_{ti} = tf_i \times idf_i \quad (3)$$

$$idf_i = lg\left(\frac{|S|}{df_i}\right) \quad (4)$$

$$Score_{keyword}(d_i,q) = \frac{rel(q|d_i)}{rel_{max}(q \mid d_i)} \quad (5)$$

式中 tf_i 为查询 q 或文档 d_i 中词 t_i 出现的频率，idf_i 为逆向查询频率，df_i 为包含 t_i 的文档数，$|S|$为样本集文档总数。

1.4.2 查询与样本集文档主题相关度

由于查询通常包含的词较短，而词的二义性普遍存在，直接利用“主题-词”分布 φ 推断其主题过于粗糙导致不够准确，从而影响检索效果。LBCS 方法使用历史查询词来扩展当前查询词，利用扩展查询来推断当前查询的主题。

由定义 1 可知，扩展查询中各词的影响度，与两个查询的相似度有关。查询 q 与历史查询 p 的相似度计算函数 sim($p|q$)需要满足以下条件：首先，sim($p|q$)的取值范围在[0, 1]；其次，两个完全相同的查询相似度为 1，即 sim($p|q$)=1；最后，两个查询的相似性越高，则 sim($p|q$)的值越大。相似度函数计算如下：

$$sim(p \mid q) = \frac{N(result(p) \cap result(q))}{N(result(p) \cup result(q))} \quad (6)$$

其中 *result*(q)和 *result*(p)分别为查询 q 和查询 p 检索样本集合得到的返回结果的文档集合。$N(result(q) \cap result(p))$表示查询 q 和查询 p 检索结果中相同的文档数，$N(result(q) \cup result(p))$表示查询 q 和查询 p 检索结果中的文档总数。

扩展查询 q'中的词 t_j 与文档 d_i 的在 LDA 模型中得出的主题$\{z_1, z_2, \ldots, z_k\}$上的主题相关度计算如下：

$$P(t_j \mid d_i) = \sum_{x=1}^{k} P(t_j \mid z_x, \varphi) \times P(z_x \mid d_i, \theta) \quad (7)$$

其中 $P(t_j \mid z_x, \varphi)$ 为通过“主题-词”分布 φ 可以得到查询 q 中的词 t_j 与主题 z_x 相关的概率。而另一个 $P(z_x \mid d_i, \theta)$ 为通过“文档-主题”分布 θ 得到的样本集中的文档 d_i 与主题 z_x 相关的概率。

通过扩展查询 q'中各个词与文档的主题相关度，可以得到扩展查询 q'与文档 d_i 的主题相关度，计算如下：

$$P(q'|d_i)=\sum_{t_j\in q'}P(t_j|d_i)\times eff(t_j) \quad (8)$$

其中 $eff(t_j)$为词 t_j对扩展查询 q'的影响度。

将之归一化得到原查询 q 与文档 d_i的主题相关度：

$$Score_{lda}(d_i,q)=\frac{P(q'|d_i)}{P_{max}(q'|d_i)} \quad (9)$$

查询 q 与文档 d 的综合相关度计算的伪代码如算法 1 所示，首先，计算查询向量 q 的模和文档向量和查询向量的内积（第 2-7 行），计算文档向量 d 的模并计算查询与文档的关键词相似度，将相似度除以相似度最大值归一化，得到查询与文档关键词相关度（第 8-12 行）。然后，通过历史查询生成原查询 q 的扩展查询 q'（第 13-20 行），计算扩展查询 q'与文档的主题相关度，并将相似度除以相似度最大值归一化，得到原查询 q 与文档的主题相关度（第 21-26 行）。最后，计算查询 q 与文档的综合相关度（第 27 行），返回各文档的综合相关度（第 29 行）。

算法1：*docsScore*(*q*)

输入：查询语句*q*

输出：样本集中文档与查询的综合相关度

```
1：FOR each document d in the sample collection
2：  FOR each term t in q
3：    normQ = getWeigth(q, t) ×getWeigth(q, t);
4：    IF( d.contains(t) ) Then
5：      dotProduct += getWeight(d, t) ×getWeight(q, t);
6：    END IF
7：  END FOR
8：  FOR each term t in q
9：    normD = getWeigth(d, t) ×getWeigth(d, t);
10：   termSim = dotProduct / sqrt(normQ+normP);
11：   score_keyword = termSim / max(termSim);
12：END FOR
13：Expand query q' = Ø;
14：FOR each past query p in P
15：  sim(p|q) = same_in_result(p, q) / total_in_result(p, q);
16：  FOR each term t_d in p
17：    q' ∨ = t_d;
18：    weight(t_d|q') = sim(p|q);
19：  END FOR
20：END FOR
21：FOR each topic z in LDA model
22：  FOR each term t' in q'
23：    p_qd += getProbability(t', z, φ) ×getProbability(z, d, θ) ×weight(t'|q');
24：    score_lda = p_qd / max(p_qd);
25：  END FOR
26：END FOR
27：scorelist[d] = λ ×score_lda + (1-λ) ×score_keyword;
28：END FOR
29：RETURN scorelist[];
```

1.5 查询与集合相关度

将样本集里的各文档，按照与查询相关度从高到低排序，对于采样自不同集合中的两个文档，显然与查询相关度更高、排名更靠前的文档所在集合与查询的相关度更高。同时，在样本集的检索结果中，属于某集合的文档数量越多，则该集合对查询的贡献度越大，集合相关度就越高。

检索结果中的每个文档对其所在集合相关度的相关度权重：

$$R(d_k|q)=\begin{cases}Score(d_k,q)\times\sqrt{1-k/\gamma} & k\le\gamma\\ 0 & k>\gamma\end{cases} \quad (10)$$

$$\gamma=ratio\times|S| \quad (11)$$

其中 d_k 为样本集中与查询的综合相关度排名为 k 的文档，$Score(d_k, q)$为文档 d_k与查询 q 的综合相关度，$|S|$为样本集的文档总数，γ 为样本集中与查询相关文档的阈值，表示与查询相关的有效文档数量，参数 *ratio* 表示检索结果的相关文档数占样本集文档总数的比例。

集合的大小也是评价集合相关度的重要因素。假设两个集合在样本集 top γ 个检索结果中拥有相同数量文档，由于基于查询的采样每个集合返回的被采样文档的总数目几乎相同，那么显然更大的集合可能包含有更多的相关文档。综合考虑上述因素，LBCS 按如下的方式估计集合 C_i 与查询的相关度：

$$Relevance(C_i)=\frac{|Ci|}{|S_{ci}|}\sum_{d_k\in S_{ci}}R(d_k|q) \quad (12)$$

其中$|C_i|$为集合 C_i包含的文档总数，S_{ci}为样本集中采样自 C_i的文档，$|S_{ci}|$为样本集中采样自 C_i的文档总数。

查询与集合相关度计算的伪代码如算法 2 所示，通过算法 1 得到样本集中文档与查询的综合相关度（第 1 行），对样本集中文档按其与查询的综合相关度从高到低排序得到排序列表（第 2 行）。选择排名大于阈值 γ 的文档（第 4 行），计算该文档对其所在集合与查询的相关度的影响（第 5 行），计算集合与查询的相关度（第 6、7 行），并返回各集合的相关度的得分（第 9 行）。

算法2：colRelevance (*q*)

输入：查询语句*q*

输出：各集合与查询的综合相关度

```
1： scorelist = docsScore(q);
2： ranklist = Rank( scorelist );
3： FOR each document d ranked k in ranklist
4：   IF ( ( k < γ ) AND d is in collection c Then
5：     R_d = scorelist[d] ×sqrt(1 – k / γ);
```

6： *collist*[*c*] += *R_d* × *NumOfDocs*[*c*] / *NumOfDoc*[*c_sample*];
7： END IF
8： END FOR
9： RETURN *collist*[];

2 实验

2.1 测试集

实验使用的测试集为 TREC 数据集 The ClueWeb09 Dataset Category B，包含有五千多万个英文网页，压缩后大小为 229.9G。实验中，数据集被随机划分成 100 个大小不等的集合，这些集合中包含的文档数最多为 123.3 万篇，最少为 14.3 万篇，平均为 50.2 万篇，压缩后集合大小，最大为 6.25GB，最小为 821MB，平均为 2.29GB。

TREC 给出了基于该测试集得 50 个真实查询话题（TREC topic1-50）以及每个查询所对应的相关文档。

2.2 评价指标

R_m（召回率），*P@n*（前 *n* 个检索结果的准确率）以及平均准确率 *MAP* 通常被用作集合选择方法的作为评价指标。

$$R_m = \frac{\sum_{i=1}^{m} E_i}{\sum_{i=1}^{m} B_i} \quad (13)$$

其中，E_i 为测试的集合选择方法选择的第 *i* 个集合中包含的相关文档的总数，B_i 为理想的集合选择方法选择的第 *i* 个集合（理想的集合选择方法按集合中相关文档总数从多到少排序）中包含的相关文档的总数，R_m 的值越大，说明被测的集合选择方法选出的集合序列越接近理想的集合排序。*P@n* 表示在集合选择方法选出的集合中检索，返回前 *n* 个结果的准确度，*P@n* 的值越大，则检索的准确度越高。本文中，*Rm* 与 *P@n* 都是 50 个查询的平均值。

2.3 实验设置

2.3.1 LBCS 参数设置

本文的 LBCS 方法包含两个参数，λ 和 *ratio*。其中 λ 为主题相关度得分在综合相关度得分的权重，λ 越大表示主题相关度得分在综合相关度得分中影响更大。实验中设置 λ=0.3，后面的实验结果也证明了 λ 取值 0.3 将获得不错的检索效果。参数 *ratio* 表示检索结果的相关文档数占样本集（所有集合）文档总数的比例，*ratio* 过大将会引入许多不相关文档，*ratio* 过小可能会过滤掉许多相关文档。Callan 指出[6]在基于查询的采样中，样本集中相关文档数占样本集文档总数的比例取值在 0.002-0.005。实验中设置 *ratio*=0.003。

2.3.2 LDA 参数设置

在通过基于查询的采样方法对各集合采样构成样本集合后，需要对样本集合中的文档建立 LDA 主题模型。按照研究中的通常设定[15]，设置 α=50/*K*，β=0.01，实验显示 α 和 β 的值对最终检索效果的影响很小。另外需要确定的参数是 LDA 的迭代次数 *Iterations* 和主题数目 *K*。采用吉布斯抽样的方法推断 LDA 的分布，迭代次数 *Iterations* 越大，推断出的分布越接近真实分布，但同时需要的时间也更长。图 3 显示了当 *K*=400，*M*=5，*n*=10 时，*Iterations* 对检索结果准确度的影响，当 *Iterations* 小于 170 时，检索结果的准确率随着迭代次数 *Iterations* 的增加而显著的提高，而当 *Iterations* 大于 170 时，检索结果的准确率趋于平稳，本文选取 *Iterations*=200。

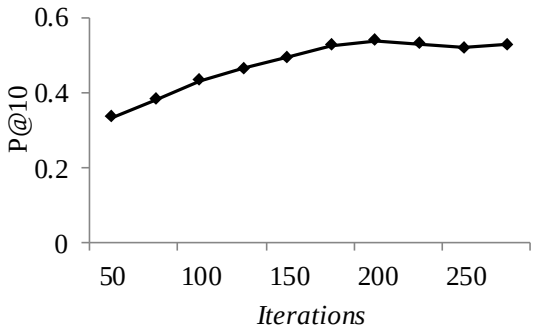

图 3 当 *K*=400，*M*=5，*n*=10 时，不同 Iterations 值对检索准确度的影响

选取合适的主题数目对建立的主题模型的优劣同样有着重要的影响。主题数过小不能充分的反应文档集合的主题，主题数过多建模时耗时越长。图 4 显示了当 *M*=5，*n*=10 时，主题数的多少对检索准确度的影响，当 *K* 到达一定数目时检索结果的准确率趋于平稳，实验证明取 *K*=400 能得到很好的检索效果。

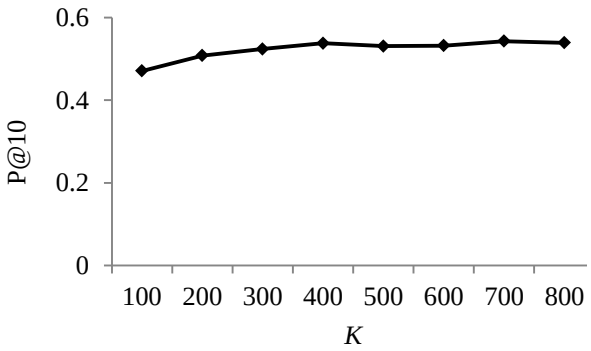

图 4 当 *M*=5，*n*=10 时，不同主题数对检索结果的影响

2.3.3 基准方法

ReDDE 方法作为基准方法广泛的用于集合选择方法的研究，因而被本文选作基准的对比方法之一。由于 CRCS(e)方法的效果显著的好于 CRCS(l)[7]，因而选择 CRCS(e)方法作为基准的对比方法之一。在对比方法中，参数均按照原文中的设置，ReDDE 方法的参数相关文档占样本集文档的比例取 0.003，CRCS(e)方法的两个参数分别取 1.2 和 2.8。

2.3.4 采样参数设置

实验采用基于查询的采样方法[16]获取各集合的描述信息。从查询日志随机选取 1 个查询作为初始查询词；每轮检索中，每个集合返回其检索结果的前 3 个文档加入到样本集合，并从样本集文档中随机抽取 1 个关键词作为下一轮检索的关键词。当样本集合中的文档数量大于或等于所有集合总文档数的 3%时，即停止采样。

2.3.5 合并策略

实验中采用两次往返的查询方式的检索合并策略。第一次统计全局信息；第二次将查询与全局信息发送到要检索的集合，以计算出文档的全局评分，使各文档的评分具有可比性。对于 top *n* 的查询，被检索的集合返回 *n* 个检索结果，检索代理服务器合并各集合返回的结果。

2.4 实验结果与分析

图 5 显示当分别选择 1 到 10 个集合时，ReDDE、CRCS(e)和 LBCS 三种集合选择方法在这 50 个查询下的平均召回率 *Rm* 值。在分别选择 1 到 10 个集合时，LBCS 方法在每个集合数的召回率的值都高于 ReDDE 方法和 CRCS(e)方法，较 ReDDE 方法和 CRCS(e)方法分别平均提高 14.2%和 19.9%，当选择的集合数增大的时候，LBCS 的 Rm 值增长更快。

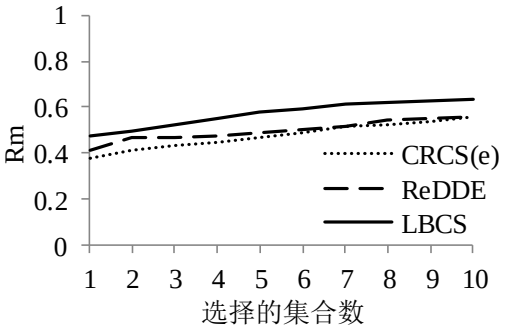

图 5 三种集合选择方法的召回率

实验结果表明，LBCS 相较于 ReDDE、CRCS(e)有更高的召回率，说明基于 LDA 主题模型的集合选择方法能有效地挖掘出查询和集合的语义层面的关系，使得 LBCS 能更准确的选择到包含更多相关文档的集合。

表 1-表 4 显示了 ReDDE、CRCS(e)和 LBCS 三种集合选择方法返回 *n* 个检索结果的准确率 *P@n* 结果对比，当三种集合选择方法分别选择相关度最高的 2，5，7，10 个集合进行检索时，50 个查询分别返回 5，10，15，20，30，40，50 个结果的平均准确率。当选择 2，5，7，10 个集合进行检索时，LBCS 较 ReDDE 准确率分别平均提高 7.8%，8.8%，5.6%，4.8%，LBCS 较 CRCS(e)准确率分别平均提高 8.7%，10.5%，6.8%，7.0%。结果表明，LBCS 相比于 ReDDE 和 CRCS(e)在选择相同集合数目时有更高的准确率，说明 LBCS 相比于 ReDDE 和 CRCS(e)能够选择到与查询更相关的集合。

表 1 选择 2 个集合时两个方法的准确率

P@n 平均值	ReDDE	CRCS(e)	LBCS
n=5	0.503	0.501	0.544
n=10	0.448	0.451	0.517
n=15	0.392	0.376	0.426
n=20	0.318	0.323	0.351
n=30	0.263	0.281	0.278
n=40	0.222	0.214	0.229
n=50	0.186	0.175	0.192

表 2 选择 5 个集合时两个方法的准确率

P@n 平均值	ReDDE	CRCS(e)	LBCS
n=5	0.532	0.525	0.564
n=10	0.516	0.498	0.538
n=15	0.483	0.484	0.522
n=20	0.456	0.446	0.501
n=30	0.423	0.406	0.466
n=40	0.386	0.382	0.425
n=50	0.351	0.357	0.399

表 3 选择 7 个集合时两个方法的准确率

P@n 平均值	ReDDE	CRCS(e)	LBCS
n=5	0.584	0.589	0.603
n=10	0.571	0.571	0.591
n=15	0.553	0.548	0.577
n=20	0.534	0.523	0.556
n=30	0.501	0.487	0.531
n=40	0.475	0.461	0.510
n=50	0.441	0.443	0.488

表 4 选择 10 个集合时两个方法的准确率

P@n 平均值	ReDDE	CRCS(e)	LBCS
n=5	0.638	0.616	0.665
n=10	0.611	0.603	0.647
n=15	0.599	0.594	0.621
n=20	0.582	0.573	0.601
n=30	0.543	0.532	0.573
n=40	0.514	0.496	0.541
n=50	0.483	0.476	0.512

另外可以看到，当返回相同数量的检索结果时，选择的集合数目越多，三种方法的准确率越高。当选择同样数目的集合时，返回的结果数 *n* 值越大，即返回的检索结果越多，三种方法的准确率越低，说明随着返回的检索结果数目变大，更多的不相关文档会混入其中，并且当选择的集合越少时，随着返回结果数的增加，混入的不相关文档比例更大。

图 6 显示了对于 50 个查询（TREC topic1-50），CRCS(e)、REDDE 和 LBCS 分别在选择 2，5，7，10 个集合时前 10 个结果的 MAP 值。当选择 2，5，7，10 个集合进行检索时，LBCS 较 ReDDE 的 MAP 值分别平均提高 9.2%，5.1%，3.7%，2.9%，LBCS 较 CRCS(e)的 MAP 值分别平均提高 8.2%，7.7%，4.7%，3.8%。结果表明，LBCS 相比于 ReDDE 和 CRCS(e)在选择相同集合数目时有更高的 MAP 值，说明 LBCS 相比于 ReDDE 和 CRCS(e)在得到相同的召回率时，检索结果的准确度更高。

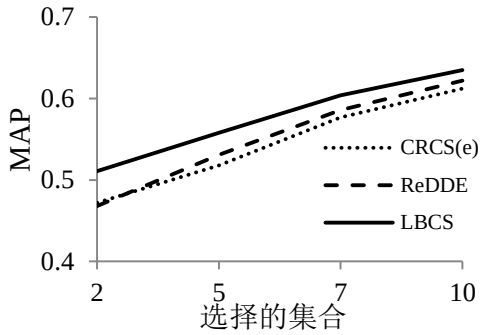

图 6 三种集合选择方法的 *MAP*

3 结语

集合选择是分布式信息检索系统中的重要环节，可有效减少网络带宽消耗，减少检索时的计算开销，提高分布式信息检索系统的效率。本文提出基于 LDA 主题模型的集合选择方法 LBCS，引入 LDA 方法对样本集建立主题模型。实验结果表明，LDA 主题模型能有效挖掘查询与集合潜在的语义关系，从而选择到与查询更相关的集合，有效提高了最终检索结果的准确率及召回率。

本集合选择方法主要以提高检索准确率与召回率为目标文设计的，然而在实际应用中，可能需要考虑更加复杂的因素，如检索服务器的吞吐率，负载，响应时间，各检索服务器的检索策略等。若能充分考虑 DIR 系统的性能以及检索的准确度与召回率，建立统一的集合选择模型将是一个很好的研究方向。

参 考 文 献

[1] Callan J. Distributed information retrieval. Croft W B. Advances in information retrieval. USA: Kluwer Academic Publishes, 2000: 127-150.

[2] Callan J P, Lu Z, Croft W B. Searching distributed collections with inference network[C]//Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1995: 21-28.

[3] Xu J, Croft W B. Cluster-based language models for distributed retrieval[C]//Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1999: 254-261.

[4] Si L, Jin R, Allan J. et al. A language modeling framework for resource selection and results merging[C]//Proceeding of ACM Conference on Information and Knowledge Management. McLean, Virginia, USA, 2002: 391-397.

[5] Yuwono B, Lee D L. Server Ranking for Distributed Text Retrieval Systems on the Internet[C]//DASFAA. 1997, 97: 41-49.

[6] Si L, Callan J. Relevant document distribution estimation method for resource selection[C]// Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003: 298-305.

[7] Shokouhi M. Central-rank-based collection selection in uncooperative distributed information retrieval[M]//Advances in Information Retrieval. Springer Berlin Heidelberg, 2007: 160-172.

[8] Thomas P, Shokouhi M. SUSHI: scoring scaled samples for server selection[C]//Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009: 419-426.

[9] Kulkarni A, Tigelaar A S, Hiemstra D, et al. Shard ranking and cutoff estimation for topically partitioned collections[C]//Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, 2012: 555-564.

[10] Cetintas S, Si L, Yuan H. Learning from past queries for resource selection[C]//Proceedings of the 18th ACM conference on Information and knowledge management. ACM, 2009: 1867-1870.

[11] 刘颖, 陈岭, 陈根才. 基于历史点击数据的集合选择方法[J]. 浙江大学学报 (工学版), 2013, 47(1): 23-28.

[12] Wauer M, Schuster D, Schill A. Integrating explicit semantic analysis for ontology-based resource selection[C]//Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services. ACM, 2011: 519-522.
[13] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. the Journal of machine Learning research, 2003, 3: 993-1022.
[14] Porteous I, Newman D, Ihler A, et al. Fast collapsed gibbs sampling for latent dirichlet allocation[C]//Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008: 569-577.
[15] Wei X, Croft W B. LDA-based document models for ad-hoc retrieval[C]//Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2006: 178-185.
[16] Callan J, Connell M. Query-based sampling of text databases[J]. ACM Transactions on Information Systems (TOIS), 2001, 19(2): 97-130.