基于检索结果聚类的XML伪相关文档查找

钟敏娟 万常选 刘德喜 廖述梅
(江西财经大学信息管理学院 南昌 330013)
(江西财经大学数据与知识工程江西省高校重点实验室 南昌 330013)

摘 要 传统伪相关反馈容易产生"查询主题漂移",有效避免"查询主题漂移"的首要前提是确定高质量的相关文档,形成与用户查询需求相关的伪相关文档集合。在检索结果聚类的基础上,研究了XML伪相关文档查找方法,在充分考虑XML内容和结构特征的前提下,提出了基于均衡化权值的簇标签提取方法,并以此为基础,提出了候选簇的排序模型和基于候选簇的文档排序模型。相关实验数据表明,与初始检索结果相比,排序模型获得了较好的性能,有效地查找到了更多的XML伪相关文档。

关键词 信息检索,XML伪相关反馈,XML检索结果聚类,簇标签,排序模型
中图法分类号 TP391 **文献标识码** A

Finding XML Pseudo-relevance Document Based on Search Results Clustering

ZHONG Min-juan WAN Chang-xuan LIU De-xi LIAO Shu-mei
(School of Information Technology, Jiangxi University of Finance and Economics, Nanchang 330013, China)
(Jiangxi Key Laboratory of Data and Knowledge Engineering, Jiangxi University of Finance and Economics, Nanchang 330013, China)

Abstract Recently study shows that traditional pseudo-relevance feedback may bring topic drift. Therefore, to avoid topic drift effectively, it is essential to identify relevant documents and to form the pseudo relevant documents to user's query. In this paper, based on clustering XML search results, a method was proposed to find good feedback documents. Firstly, a cluster-label extraction method based on equalizing weights was introduced, by fully considering the content and structure features in XML documents. Secondly, a two-stage ranking strategy was presented, as the candidate cluster ranking model and document ranking model. Finally, experimental data shows that compared to original retrieving method, the ranking models obtain better performance and find more relevant XML documents.

Keywords Information retrieval, XML pseudo-relevance feedback, XML search results clustering, Cluster label, Ranking model

1 引言

XML文档是具有层次结构和文本内容的半结构化数据。随着网络应用的快速发展,符合XML规范的数据(称为XML数据)成为网络上信息描述和信息交换的事实标准,并大量存在于当前的信息社会,尤其是电子商务、Web服务、数字图书馆等应用领域。如何有效地获得高质量的相关信息已成为信息检索领域亟待解决的一个重要问题。

近年来,已有许多针对XML文档信息检索技术的研究,其中,利用伪相关反馈技术,帮助用户形成更准确的查询表达式,从而进一步提高XML检索效果,成为一个重要的研究课题。伪相关反馈容易产生"查询主题漂移"现象,最主要的原因是传统伪相关反馈需要事先固定N值,并且假定这N篇文档就是相关的。而研究表明[1],不同的N值对最终检索性能有较大的影响。从实际情况考虑,前N篇文档并不总是与查询相关,不同的查询主题所获得的初始检索结果质量是不同的。有些查询可能在初始检索结果的前N篇文档中包含较多的相关文档,质量较高;有些查询可能恰恰相反,初始的结果里存在大量无关文档,从不相关的文档中提取扩展信息显然不合适。因此有效避免"查询主题漂移"的首要前提是确定高质量的相关文档,形成与用户查询需求相关的伪相关文档集合。

文献[2]提出了一系列的特征,比如查询词项的熵、文档的初始相关度值、反馈文档与整个反馈文档集的相似度以及扩展词与初始查询词之间的距离等因素,并根据这些特征将高质量的相关文档从反馈文档集里过滤出来。文献[3]提出了伪无关文档(Pseudo-Irrelevant Document)的概念。顾名思义,伪无关文档就是与用户查询不相关的文档,令F_R代表初始检索结果的前N篇文档集,F_I表示伪无关文档集,X是超出N之外的高分值文档集,Y是与F_R集合中任意文档相似

到稿日期:2012-12-12 返修日期:2013-03-19 本文受国家自然基金项目(61173146,61262035,60763001),国家社会科学基金(12CTQ042)资助。

钟敏娟(1976—),女,博士,讲师,CCF会员,主要研究方向为XML信息检索、智能信息处理;万常选(1962—),男,博士,教授,主要研究方向为Web数据管理、XML数据库、信息检索、数据挖掘;刘德喜(1975—),男,博士,副教授,主要研究方向为信息检索、自动文摘、智能信息处理;廖述梅(1976—),女,博士,副教授,主要研究方向为数据挖掘。

的文档集，则 $F_I=X-(X\cap Y)$。通过在初始检索结果集里去除伪无关文档，保证查询扩展能在相关文档中进行。文献[4]提出了一个学习算法 FeedbackBoost，该方法基于 Boosting 框架，能针对不同的查询主题获得较高质量的反馈文档以及扩展词，从而提高整个伪相关反馈过程的健壮性和有效性。

对初始检索结果聚类是确定相关文档的另一种有效手段。通过聚类对初始检索结果集进行取样与重取样，从而提高伪相关文档质量。

文献[5]提出了基于取样的伪相关文档选择标准，对前 N 个返回文档进行筛选。筛选的原则体现了聚类的思想，即文档之间如果包含的词项重复过多，则这些文档应该被聚为一个簇，同时，这样的相似文档不应该被重复取样而选入反馈文档中。该方法的最大局限就是一些不相关的文档会引入到伪相关文档集里，从而带来噪音。文献[6]提出了基于聚类分析的重新取样方法来更好地选择伪相关文档，这个更好的伪相关文档就是领域文档(dominant document)。文中采用 K-nearest neighbors(K-NN)方法对检索结果的前 N 篇文档进行可重叠聚类。该方法的不足之处是在一些应用场景，比如文档长度和文档中词项的表达方式存在很大不同的数据集里效果不显著。因此，在此基础上，文献[7]提出了一种改进的聚类过程用于伪相关反馈，该方法基于局部频率词项检查簇的内部相似度并进行相似簇的合并，然后对各簇进行排序，那些分布在位序较前的簇里且同时与簇中心局部频率词项的相似性高的文档被选作伪相关文档。Collins&Callan 在文献[8]中提出了不确定性是信息检索的特性，比如在查询意图表达、文档语义表达等方面均存在着不确定性。文中采用不确定性分析的取样方法来构造反馈模型，引入了 bootstrap 方法来对初始检索结果的前 N 个文档进行重抽样，并对查询进行不同的变形，以达到提高检索的准确性与稳定性的目的。

虽然较国外起步较晚，但国内在这部分的研究也取得了不少成果。叶正[9]在博士论文中提出了一个质量偏重模型，即通过引入一个质量因子来衡量反馈文档的质量，从中将高质量的反馈文档选择出来。其中文档质量由多种因素决定，包括文档长度、词项特征、Bigram 特征以及查询词间平均距离等。文献[10]在统计语言模型的框架下提出了一种基于独立分量分析(Independent Component Analysis，ICA)的统计语义聚类方法，文中假设文档由多个隐含的独立主题混合噪声组成，ICA 可以分离文档中的独立主题形成 ICA 语义空间，并对此进行软聚类。通过用户初始查询与语义聚类的相似度决定使用某个语义聚类下文档估计查询模型。该模型由于滤去了语义聚类下查询不相似文档，因此反馈文档质量得到了提高。文献[11]提出了一种基于相关反馈的算法，即通过一定的基于反馈结果的评价方式来预测反馈文档的相关性，从中选取最优的文档。

以上这些方法都是基于传统文本的伪相关反馈，并没有考虑 XML 的结构特征。

本文的主要贡献有：

(1)提出了基于检索结果聚类的 XML 伪相关文档查找技术路线。该技术路线区别于现有的方法，不仅能通过聚类将主题相似的文档聚簇在一起，而且在此基础上能充分利用聚类结果的相应特征来挑选好的伪相关文档。具体表现在：在候选簇的排序模型中，能充分利用簇中心与查询的相似性获得较好的候选相关簇；在基于候选簇的相关文档排序模型中，依据文档与簇的相似性、簇的排名等特征将排序在前的候选簇中的相关文档选取出来。实验数据表明该思路是行之有效的，一方面结合聚类特征获取了较好质量的伪相关文档集，有效地解决了传统伪相关反馈中扩展源质量不高的问题；另一方面也用实验验证了聚类在整个伪相关文档查找中所起的作用，能有效地帮助查找到更多相关的 XML 伪相关文档集。

(2)提出了基于均衡化权值的簇标签提取方法，该方法能充分考虑 XML 文档内容和结构的双重特点，将反映簇主旨思想的中心词项挑选出来。

(3)在检索结果聚类的基础上提出了两阶段的排序模型，即候选簇的排序模型和基于候选簇的相关文档排序模型。通过两阶段的排序，该模型获得了较高质量的 XML 伪相关文档集，有效地确保了扩展源的质量。

2 基于均衡化权值的簇标签提取

从簇中提取最具代表性的信息即簇标签来描述簇能帮助用户迅速确认生成的文档类相关与否，并且通过浏览簇标签知道各个组/簇的主题。本文拟采取词项的形式，将每个簇的中心词项挑选出来作为最终簇的标签。簇是由多篇文档构成的，因此簇的中心词项应该自然而然地由文档的中心词项汇聚而成。通常，词项权重值反映了词项在文档中的重要程度。词项越重要，相应的权值也越大。选择较高权值的词项能在一定程度上代表该篇文档，反映该篇文档的主旨思想。因此，文本利用词项权重特征选择文档候选中心词项。词项权重采用文献[12]中式(1)和式(2)计算得到。

在此基础上进行簇中心词项的选取。选取簇中心词项需考虑如下特征：

(1)候选中心词项的 DF(document frequent)特征：DF 是指一个簇中以该词项作为中心词项的文档个数。把 DF 作为簇的中心词项的一个统计指标是基于以下假设：如果一个候选词项在同一个簇的很多篇文档里都作为中心词项，那么该候选词项具有比较广泛的代表性，因而可以作为该类别的标签。例如，查询词“code signing”，我们发现簇里的很多文档的中心词项都出现了 java、system，说明该词项代表了很多篇文档的中心思想，因此可以挑选其作为该簇的中心词项。

(2)簇中候选中心词项的平均相对排名。候选中心词项在每篇文档中都有排名，有些词项在某篇文档中排序较前。在其他篇文档中排序较后，因此可以利用该词项的平均相对排序值进行选择，平均相对排序值比较靠前，说明该词项在簇中的大多数文档中都排列较前，对大多数文档而言，该候选中心词项对这些文档的贡献程度也比较大，因此可将此作为挑选准则。

具体计算公式如下所示：

$$Aver\_RankValue(term_i)=\frac{\sum_{i=1}^{count} Rank\_Value(term_i,d_j)}{count} \quad (1)$$

式中，$count$ 表示一个簇中包含文档的篇数，$Rank\_Value(term_i,d_j)$ 表示词项 $term_i$ 在文档 d_j 中的排名，计算如下：

$$Rank\_Value(term,d_i)=(Top\_N-position_i-1)/Top\_N \quad (2)$$

定义均衡化权值为候选中心词项的 DF 特征值与相应的平均相对排名的乘积，计算如下：

$$balance\_weight(term_i) = DF(term_i) * Aver\_RankValue(term_i) \quad (3)$$

中心词项的最终权值计算如下：

$$W(t_i, c_j) = balance\_weight(t_i) * \ln(\frac{N+1}{n_{t_i}}) \quad (4)$$

式中，N 表示整个数据集划分的簇数，n_{t_i} 表示所有簇中以词项 t_i 作为中心词项的簇的个数。

3 两阶段的排序模型

查找高质量的伪相关文档其实质就是排序问题，它包含两个阶段的任务：第一阶段，在检索结果聚类的基础上，根据查询条件与簇的相似程度，对各个簇进行相似度排序，挑选出较为相关的候选簇，其中检索结果聚类采用文献[9]中的思路；第二阶段，在选定的相关簇中，利用与查询条件的相似性，对簇中的文档进行排序，排序越前，说明相似程度越高，质量越好。通过这两个阶段，将排序靠前的 N 个文档形成伪相关文档集合。

3.1 候选簇的排序模型

当完成检索结果聚类后，为了获取较高质量的伪相关文档集，需要对聚类结果进行后续的分析处理。聚类过程完成了区分相关结果与不相关结果的工作，从理论上讲，聚类的最佳效果应该达到将相关文档完全归到一个簇中，而不相关的结果归到其它簇中，但实际情况，相关文档可能分布在多个簇。因此，为了能够更好地找到相关文档，避免查全率的降低，我们需要选择多个候选簇，而不仅仅是单个簇。考察一个簇为合适的候选簇，我们认为，它应该满足如下条件：

条件 1　簇中应该包含有相关文档；

条件 2　相比其他簇而言，该簇包含有更多的相关文档；

条件 3　在簇的文档数目一定的前提下，该簇的相关文档在整个簇中所占的比值越高越好。

从以上条件可以分析，对一个较好的候选簇而言，簇内分布的文档与用户查询意图的关联度应该是比较高的。理想状态下，如果某个簇中所有的文档都与用户的查询相关，那么这个簇应该是相似程度最高的簇，也理应被选为候选簇。因此，很自然地需要依次考察每个簇中的每篇文档与查询的相关程度，假如相关程度高，则该簇可以被选为候选簇。然而，这种方法显然使得执行效率过于低下，特别是聚类的文档数目较多的情况。因此，应该选用簇的公共特征来衡量一个簇是否符合候选相关簇的条件。簇的中心点在一定程度上能够表征整个簇，因此，本文利用查询与簇中心的相似性对簇进行选择，将与查询相关度高的簇挑选出来。k-medoid 聚类算法最终会产生 k 个中心点，利用这 k 个中心点文档与查询的相似性，将相似程度高的簇挑选出来，即为候选的相关簇。相似性计算基于文献[13]中的调整，公式如下：

$$\mathrm{sim}(d_i, Q) = \sum_{t_k \in Q} \frac{1+\ln(tf(t_k, d_i)+1)}{(1-s)+s\frac{dl}{avdl}} \times tf(t_k, Q) \times IEF(t_k)$$

$$IEF(t_k) = 1 + \log\frac{N}{N_k} \quad (5)$$

$$tf(t_k, d_i) = \sum_{j=1}^{m} \frac{w(\sigma_{ij})}{Level(\sigma_{ij})^2} * (\ln(tf(t_k, \sigma_{ij}) + 1))$$

式中，$tf(t_k, Q)$ 表示词项 t_k 在查询 Q 中的出现频率，$tf(t_k, d_i)$ 表示词项 t_k 在簇中心文档 d_i 中的出现频率，s 是一个试验参数（通常取 0.2），dl 是簇中心文档 d_i 的长度，$avdl$ 是数据集中文档的平均长度，$IEF(t_k)$ 是词项 t_k 的反比元素频率。

对各个簇进行相似度排序，排在前 N 的簇号即为候选的相关簇。

3.2 基于候选簇的文档排序模型

候选的相关簇确定之后，接下来需要在候选簇中选择高质量的反馈文档，其实质就是对候选簇中的所有文档进行相似性排序。我们认为排序在前的文档与用户的查询意图相关程度高，因此，需要利用一些特征来辅助相关文档的选取，过滤掉低质量的反馈文档。本文主要考虑以下特征：

(1)相关度值(R_Score)：在信息检索领域，相关度值代表着与查询主题的相关程度，值越高，与用户的查询越相关，在结果文档排序中越靠前排列，从而也更有机会成为较高质量的反馈文档。

(2)与簇的相似性(Cluster_Sim)：一篇候选文档是不是相关文档，除了要考虑该文档自身与查询主题的相关程度以外，还要考虑该文档与整个簇的相关性。事实上，候选相关簇中相关文档与不相关文档混杂在一起，因此，候选簇中的文档并不一定就完全与主题相关，为此，要把这些不相关的文档噪音过滤，或者让它们排序在后。根据前面候选簇的选取准则，候选簇应该是与用户的查询主题最为相关的，因此，可以将候选文档与所在簇的相似性作为一个参照标准，来间接地衡量候选文档与查询的相似程度。

(3)所在簇的相对排名(Cluster_RValue)：候选文档位于候选簇中，簇的排名体现了该簇与查询条件的相关程度，排名越前，与查询的关联程度越高。因此，利用候选文档所在的候选簇的相对排名能间接地反映与查询的相似程度。候选文档所在的候选簇排名越靠前，说明该簇内部的对象整体上与用户的查询意图相关程度越高，因而，该候选文档成为高质量的反馈文档的概率越大。

综合以上分析，我们定义相关伪相关反馈文档的评价公式为如下形式：

$$Final\_Sim(d_i, Q) = Cluster\_RValue_i * (\alpha * R\_Score(d_i, Q) + \beta * Cluster\_Sim(d_i, c_i)) \quad (6)$$

$$\alpha + \beta = 1$$

式中，$Cluster\_RValue_i$ 表示文档 d_i 所在簇的相对排名，公式定义如下：

$$Cluster\_RValue_i = (ClusterNum - R_i)/ClusterNum \quad (7)$$

式中，$ClusterNum$ 代表聚类结果中簇的总个数，R_i 表示文档 d_i 所在簇在所有簇中的排名位序。$R\_Score(d_i, Q)$ 表示文档 d_i 与用户查询 Q 的相似度。通常，文档与查询的相似程度大都采用向量空间模型中的余弦公式，但是在这里，我们采用基于 PNW 模型的相似度计算方法，具体公式如式(5)所示。同时，$Cluster\_Sim(d_i, c_i)$ 表示文档 d_i 与所在簇 c_i 的相似度，它表征的是该文档与所在簇主旨思想的相关程度，因此，我们采用簇标签来代表整个簇。簇标签和文档均可看成由中心词项

所构成的向量。例如，对于有 n 个不同中心词项的系统，簇标签 c_i 可以表示为 $c_i=(t_{i1},t_{i2},\cdots,t_{in})$，文档 d_j 也可以表示为 $d_j=(t_{j1},t_{j2},\cdots,t_{jn})$，利用两者之间的距离可以衡量彼此的相似度。距离采用向量空间模型中的余弦公式。

4 实验分析与评价

本实验在文献[12]检索结果聚类的基础上验证所提的两个排序模型，即候选簇的排序模型以及候选簇中相关文档的排序准则能否有效地帮助查找到更多高质量的相关反馈文档。实验从两个方面来开展，一是在相同文档数目的前提下对比聚类前后高质量反馈文档的查找精度，精度越高，说明查找到的高质量反馈文档数越多，这包括验证候选簇的排序模型以及候选簇中相关文档的排序模型的有效性；二是分析聚类在查找高质量反馈文档中所起的作用，通过检索结果聚类能否帮助找到更多的高质量反馈文档。

4.1 实验步骤

具体实验步骤如下：

Step1 检索结果聚类。依据文献[12]方法，对初始的XML检索结果进行聚类。

Step2 簇的排序。在检索结果聚类的基础上，对各个测试查询的聚类结果依照候选簇的排序准则，分别计算各个簇中心文档与查询条件的相似程度，并对此进行排序。

Step3 候选簇的选择：依据簇的排序结果，选取前 N 个簇作为候选簇。因此，我们必须确定 N 值。N 值是否取得合适，关系到后续反馈文档的查找质量。N 值过大，虽然相关文档可能都包含在了这 N 个簇里，但是不相关文档占据了大部分，从而带进了大量的噪音干扰，可能会使相关文档无法排序在前。另一方面，N 值选取过小，相关文档可能涵盖不全，这样导致了大量的伪相关文档无法查找及获取。根据文献[12]检索结果聚类的实验数据，聚类的相关簇率为0.43，也就是说，相关文档平均分布在4个簇中，从这点来看，N 值必须小于4。折中数据集里相关文档分布的查全率和查准率，我们定义 N 值大小为2，即将簇中心与查询的相似程度最大的前2个簇挑选出来作为候选的相关簇。

Step4 候选簇中的文档排序。对挑选出的2个候选簇里的文档按照前述的3个特征即相关度值、与簇的相似度以及所在簇的相对排名进行排序。对照IEEE CS2005数据集里提供的文档相似性评价，得出最终反馈文档的性能质量。

4.2 实验评价

4.2.1 候选簇的排序模型评价

首先我们验证候选簇的排序模型的有效性。通常情况下，用户认为越相关的页面排序越在前，越不相关的页面排序越后，他们只会浏览前面几屏的检索结果。为此就会出现即便是相关文档，但是由于排序很后，也可能得不到用户翻查的现象，从而影响到最终检索的性能。因此，我们的最终目的不仅是把相关的反馈文档找出，而且还要将其排序在前。而要达到更多的相关文档排序在前，首要的前提是候选簇的正确选择，只有获得了正确的相关候选簇，才有可能使得后续更多的相关文档被找到。根据候选簇的排序准则，我们得到如下的实验结果，如表1所列。

表1 候选簇的选择结果

查询编号	相关的簇号	候选簇号
202	2(4/65),9(1/3)	2,5
203	7(1/15),8(1/19),9(14/32)	9,7
205	2(2/25),3(2/26),6(11/25)	6,2
206	4(2/21),5(4/8),9(15/45)	9,4
207	3(15/21),5(32/38),10(4/12)	5,3
208	5(1/8),6(11/29),7(10/25)	7,6
209	1(13/37),4(1/2),5(26/28),6(2/10),8(1/5),9(1/6),10(1/4)	1,5
210	1(3/14),5(3/6),7(1/1),8(3/31),10(8/39)	1,5
212	4(11/20),5(2/16),8(1/1),10(21/48)	9,10
213	6(10/53)	6,2
216	1(6/23),3(1/5),7(17/49),8(1/1)	7,1
217	2(3/18),6(1/17),9(2/12),10(2/24)	2,9
218	1(2/17),4(6/24),5(3/16),6(1/9),7(2/10),8(2/10),9(2/4),10(1/5)	4,9
219	1(1/15),2(1/21),10(1/12)	2,10
221	1(4/14),4(10/45),7(5/15),9(2/2),10(1/4)	4,9
222	1(27/45),3(1/13),4(1/4),6(1/4),7(1/1),8(4/9),10(13/20)	7,1
223	1(13/25),5(5/7),6(21/30),7(7/10),10(1/6)	5,7
227	2(1/36),9(12/41)	5,9
228	3(20/47),4(2/3),5(2/11),6(3/8),8(1/2),10(4/7)	3,5
229	1(1/13),3(8/44),4(1/18),9(6/13)	9,4
230	1(1/2),2(1/24),5(1/5),8(5/48)	1,2
232	3(9/24),6(1/4),9(1/32)	3,6
233	1(1/3),4(1/7),7(1/21)	7,10
234	2(1/6),3(37/42),4(10/12),6(2/7),10(1/12)	1,3
235	2(11/16),3(1/8),4(8/16),5(4/7),6(5/8),8(2/19),9(1/10),10(6/22)	2,4
236	1(17/49),3(3/4),6(1/2),7(4/7),10(1/3)	1,4
237	1(5/10),2(3/4),5(6/16),6(2/6),7(3/18),8(9/26),9(4/6),10(8/11)	10,8
239	1(13/21),5(2/7),6(2/23),10(1/2)	1,3
241	1(1/29),3(3/13),8(1/13)	3,5

其中，相关的簇号表示聚类结果里相关文档在各个簇中的分布情况，我们采取 $C_i(R\_DN_i/DN_i)$ 形式来说明。C_i 代表簇号，R_DN_i 表示第 i 个簇中相关文档的个数，DN_i 表示第 i 个簇中总共包含的文档个数。依据上述候选簇应满足的条件，从表1中的数据可以看出，基于簇中心文档的候选簇排序模型基本将好的候选簇号挑选出来了，具有较好的性能效果。从条件1的满足情况来看，2个候选簇至少有1个簇包含了相关文档，且大部分的候选簇里均包含了相关文档，为下一步候选簇中挑选高质量的相关文档奠定了前提基础。

4.2.2 相关文档的排序模型评价

候选簇挑选出来之后，接下来就是要对候选簇中的所有文档进行相似度排序，把低质量的、与用户查询需求无关的文档尽可能地摒弃在后面，而那些高质量的、与用户查询意图相吻合的文档尽可能排序在前，这就需要有一个好的合理的排序机制。

为此，对本文提出的相关文档排序模型进行了验证。利用式(6)、式(7)将候选簇中的所有文档进行相似度的排序，并与初始的检索结果进行对比。式(6)中对不同的相似度因素进行了权重的分配，显然，比重越大，相应的因素在整体评价体系中所起的作用越大，因而产生的效果也会有所不同，为此，依据我们的经验以及实际情况，文本首先对系数的不同取值进行测试。以文档与查询条件的相似度因素为基准，采用逐步添加法，每次变化0.1，依次进行实验评价，最终选取 $\alpha=0.5$，$\beta=0.5$ 为最优解。采用准确率指标 Prec@X 来衡量相关文档查找的准确率，实验结果如表2—表4所列。

表 2　Prec@10 的性能比较

查询编号	Prec@10		查询编号	Prec@10	
	初始结果	聚类后		初始结果	聚类后
202	0.10	0.00	222	0.80	0.90
203	0.60	0.70	223	0.50	0.90
205	0.90	0.50	227	0.20	0.20
206	0.50	0.60	228	0.80	0.90
207	0.90	0.90	229	0.50	0.60
208	0.50	0.40	230	0.30	0.20
209	0.80	0.80	232	0.80	0.50
210	0.40	0.30	233	0.00	0.00
212	0.40	0.30	234	0.40	0.40
213	0.60	0.60	235	0.90	0.70
216	0.20	0.50	236	0.60	0.60
217	0.20	0.30	237	0.80	0.60
218	0.30	0.50	239	0.70	0.80
219	0.20	0.10	241	0.20	0.30
221	0.50	0.50			
平均准确率				0.50	0.50

表 3　Prec@15 的性能比较

查询编号	Prec@15		查询编号	Prec@15	
	初始结果	聚类后		初始结果	聚类后
202	0.07	0.00	222	0.80	0.93
203	0.47	0.73	223	0.60	0.73
205	0.67	0.53	227	0.40	0.33
206	0.40	0.47	228	0.73	0.67
207	0.87	0.93	229	0.33	0.40
208	0.47	0.40	230	0.20	0.13
209	0.67	0.80	232	0.73	0.53
210	0.33	0.40	233	0.07	0.00
212	0.47	0.40	234	0.47	0.53
213	0.47	0.53	235	0.73	0.67
216	0.20	0.40	236	0.60	0.60
217	0.13	0.27	237	0.67	0.60
218	0.47	0.33	239	0.53	0.73
219	0.13	0.13	241	0.13	0.20
221	0.33	0.47			
平均准确率				0.45	0.48

表 4　Prec@20 的性能比较

查询编号	Prec@20		查询编号	Prec@20	
	初始结果	聚类后		初始结果	聚类后
202	0.05	0.05	222	0.80	0.80
203	0.40	0.60	223	0.45	0.60
205	0.60	0.50	227	0.35	0.35
206	0.30	0.35	228	0.60	0.65
207	0.85	0.95	229	0.35	0.30
208	0.40	0.40	230	0.20	0.10
209	0.70	0.80	232	0.55	0.45
210	0.30	0.30	233	0.15	0.00
212	0.45	0.45	234	0.50	0.65
213	0.35	0.45	235	0.65	0.70
216	0.15	0.50	236	0.55	0.60
217	0.10	0.25	237	0.60	0.55
218	0.40	0.30	239	0.45	0.60
219	0.10	0.10	241	0.10	0.15
221	0.30	0.40			
平均准确率				0.41	0.44

从上述表的数据可以明显地看出，相比初始的检索结果，该排序模型能够返回较多数目的高质量的反馈文档，并且使之排序在前，获得了较高的平均查准率，Prec@15 和 Prec@20 性能分别提高了 6.7%和 7.3%。但同时也看到在检索出的前 10 篇文档中，Prec@10 的平均精度并没有比初始的检索结果有所提高。分析原因，我们认为有以下几种可能：

(1)候选簇的选择问题。要想较多的相关文档排序在前，首先候选簇要选择正确，这意味着选择的候选簇里应该尽可能包含有较多的相关文档，并且相关文档分布率要尽可能地大。在有些测试查询里，比如查询 202，查询条件为“hidden markov model”，候选簇号为 9、10，其中，10 号簇包含了相关文档，根据候选簇的满足条件，可以选择为候选簇，但是 9 号簇里却没有包含相关文档，若这种不正确的候选簇被选择进来，就会把更多的不相关文档融入进来，带来更大的噪音，使得查找相关文档的几率在一定程度上降低。

(2)相关文档排序模型中特征的选择问题。在排序模型里，我们主要只考虑了两个方面的因素，一个是文档与查询条件的相似度，另一个是文档与所属簇的相似度，两个因素都是从查询词在文档中的频率角度来考虑，而事实上，有很多的查询并不是仅仅查询词出现得越多，文档就越相关，比如查询 202，有些文档里出现了很多的 model 词项，根据排序模型的计算，这些文档与查询条件的相似度值都比较高，因此都排列在前，但是考察它们的内容，却发现并不是关于 hidden markov 方面的 model，显然与用户的查询意图不相关，从而造成查准率比较低。因此，从多角度考虑文档与查询的相似性是值得我们进一步关注的一个问题。

4.2.3　聚类对 XML 伪相关文档查找的影响

从表 2—表 4 的数据可以看出，相比初始的检索结果，在相同文档数目的前提下，本文所提的方法能获得更多的相关文档，具有更好的性能。分析原因，本文的思路是希望借助检索结果聚类这种手段，将高质量的伪相关文档选择出来。在此，本文对聚类在反馈文档查找中的影响进行了实验分析。通过实验验证，聚类对高质量相关文档查找的影响，是否能够有利于查找到高质量的反馈文档以及更多数量的高质量反馈文档。

因此，为了衡量聚类对反馈文档的查找效果，本文对初始检索结果的前 100 篇文档重新进行了相似度计算，计算采用与聚类方法相同的准则，并同时对这 100 篇文档进行相似度排序。实验结果如表 5 所列。

表 5　Prec@10 和 Prec@20 的性能比较

查询编号	Prec@10		Prec@20	
	初始结果	聚类后	初始结果	聚类后
202	0.10	0.00	0.05	0.05
203	0.40	0.70	0.30	0.60
205	0.70	0.50	0.55	0.50
206	0.30	0.60	0.35	0.35
207	0.80	0.90	0.75	0.95
208	0.30	0.40	0.30	0.40
209	0.80	0.80	0.70	0.80
210	0.30	0.30	0.20	0.30
212	0.60	0.30	0.60	0.45
213	0.60	0.60	0.35	0.45
216	0.50	0.50	0.35	0.50
217	0.30	0.30	0.20	0.25
218	0.30	0.50	0.35	0.30
219	0.20	0.10	0.15	0.10
221	0.30	0.50	0.35	0.40
222	0.90	0.90	0.95	0.80
223	0.70	0.90	0.65	0.60
227	0.30	0.20	0.15	0.35
228	0.90	0.90	0.80	0.65
229	0.30	0.60	0.25	0.30
230	0.30	0.20	0.20	0.10
232	0.90	0.50	0.55	0.45
233	0.00	0.00	0.00	0.00
234	0.00	0.40	0.15	0.65
235	0.50	0.70	0.45	0.70
236	0.20	0.60	0.20	0.60
237	0.40	0.60	0.40	0.55
239	0.80	0.80	0.60	0.60
241	0.30	0.30	0.20	0.15
平均查准率	0.45	0.50	0.38	0.44

表5的实验数据是在相同文档数目的前提下，比较聚类前后相关文档的准确率。假如经过聚类之后，返回的相关文档数目增加了，Prec@X指标提高了，说明借助聚类能够有效地帮助查找到更多的高质量反馈文档。从表5中的数据可以看出，在相同文档数目的前提下，聚类后排在前 X 的文档里的相关文档数目要比聚类前的相关文档数目多，在Prec@10以及Prec@20指标上，平均查准率分别提高了11.1%和15.8%，因此获得了较好的性能。事实上，在相似度计算中，利用了聚类的相应特征，比如文档与簇的相似度以及文档所在簇的排名等，这些特征能有效地帮助查找到更多相关文档，并且使得它们能够尽可能地排序在前；而相反地，不进行聚类，单纯地利用文档与查询的相似性，并不能使这些相关文档排序在前。这充分说明了聚类这种手段能够有效地帮助查找到更多的高质量反馈文档，对有效避免伪相关反馈中的查询主题漂移奠定了前提基础。

结束语 确定相关文档是有效避免伪相关反馈中"查询主题漂移"的首要问题。本文在检索结果聚类的基础上，对如何查找高质量的XML反馈文档进行了研究。本文的主要工作及创新点如下：

(1)提出了基于检索结果聚类的XML伪相关文档查找方法。目前，国内外对如何确定XML相关文档的研究成果还很少，大部分的工作都是针对传统文档，借助传统的文档特征来进行。本文在综合考虑XML文档特征的基础上，结合聚类的技术手段较好地解决了XML伪相关反馈中扩展源质量不高的问题，并且用实验验证了聚类在XML伪相关文档查找中的作用。

(2)提出了基于均衡化权值的XML簇标签提取方法。该方法中词项的均衡化权值基于词项的文档权值，不仅从传统的词项频度出发，还考虑了XML文档的标签语义信息、标签的层次信息等，这些特征有效地融合了XML的内容和结构双重特性，反映了XML文档的综合语义，能有效地将反映簇主旨思想的中心词项挑选出来，并在排序模型中获得了较好的效果。

(3)为了确定伪相关文档集，在检索结果聚类的基础上进一步提出了两阶段的排序模型。候选簇的文档排序模型区别于传统信息检索排序机制，其在传统特征基础上充分考虑了聚类所带来的相应特征，比如文档与簇的相似度、候选簇的排名等。相关实验数据表明该排序模型具有一定的合理性，能获得较高质量的XML伪相关文档集。

进一步的研究工作包括：

(1)候选簇参数的自动优化工作。在候选簇的排序模型中，我们对候选簇的个数进行了固定，而在实际中，不同的簇数对高质量反馈文档的查找结果有较大的影响。因此，如何根据不同的查询主题设计参数自动寻优是今后的研究工作。

(2)排序模型中特征的提取研究。获得了好的检索结果聚类效果之后，合理的排序机制能够使相关文档被进一步查找出来并且排序在前，这需要有好的特征来体现。而文本所提的相应排序特征较为直观、简单，因此，如何更深层次、更全面地挖掘文档与查询的相关性特征，从而体现用户查询意图，也是值得关注的一个方面。

参 考 文 献

[1] Qiang H, Dawei S, Stefan R. Robust Query-Specific Pseudo Feedback Document Selection for Query Expansion[A]//Proc. of the 30th European Conf. on Information Retrieval(ECIR), 2008[C]. Heidelberg: Springer-Verlag, 2008: 547-554

[2] Ben H, Ladh O. Finding Good Feedback Documents[A]//Proc. of the 18th ACM Conf. on Information and Knowledge Management(CIKM), 2009[C]. New York: ACM Press, 2009: 2011-2014

[3] Karthik R, Raghavendra U, Pushpak B, et al. On Improving Pseudo-Relevance Feedback Using Pseudo-Irrelevant Documents [A]//Proc. of the 32nd European Conf. on Information Retrieval(ECIR), 2010[C]. Heidelberg: Springer-Verlag, 2010: 573-576

[4] Lv Yuan-hua, Zhai Cheng-xiang, Chen Wan. A Boosting Approach to Improving Pseudo-Relevance Feedback[A]//Proc. of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2011[C]. New York: ACM Press, 2011: 165-174

[5] Sakai T, Manabe T, Koyama M. Flexible Pseudo-Relevance Feedback via Selective Sampling[J]. ACM Transactions on Asian Language Information Processing, 2005, 4(2): 111-135

[6] Kyung S L, Croft W B, James A. A Cluster-Based Resampling Method for Pseudo-Relevance Feedback[A]//Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2008[C]. New York: ACM Press, 2008: 235-242

[7] Shariq B, Andreas B. Improving Retrievability of Patents with Cluster-Based Pseudo-Relevance Feedback Document Selection [A]//Proc. of the 18th ACM Conf. on Information and Knowledge Management(CIKM), 2009[C]. New York: ACM Press, 2009: 1863-1866

[8] Kevyn C T, Jamie C. Estimation and Use of Uncertainty in Pseudo-Relevance Feedback[A]//Proc. of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2007[C]. New York: ACM Press, 2007: 303-310

[9] 叶正. 基于网络挖掘与机器学习技术的相关反馈研究[D]. 大连：大连理工大学，2011

[10] 蒲强，何大庆，杨国纬. 一种基于统计语义聚类的查询语言模型估计[J]. 计算机研究与发展，2011，48(2)：224-231

[11] Gong Bi-hong, Peng Bo, Li Xiao-ming. A personalized re-ranking algorithm based on relevance feedback[A]//1st International workshop on Database Management and Applications over Networks, DBMAN, 2007[C]. 2007: 4537: 255-263

[12] 钟敏娟. 基于内容与结构语义相融合的XML检索结果聚类[J]. 情报学报，2012，31(5)：515-525

[13] Singhal A, Choi J, Hindle D, et al. AT&T at TREC-7[A]// Proc. of the 7th Text Retrieval Confernece(TREC-7), 1998[C]. NIST Special Publication, 1998: 239-252