

文章编号: 1003-0077(2007)03-0047-07

基于层次聚类的自适应信息过滤学习算法

洪宇,张宇,刘挺,郑伟,龚诚,李生

(哈尔滨工业大学 计算机科学与技术学院 信息检索实验室,黑龙江 哈尔滨 150001)

摘要: 本文采用一种基于层次聚类的自适应学习策略,从系统反馈的信息流中,动态提取一类最优信息的质心更新用户模型,有效屏蔽了阈值失真和初始信息稀疏造成的大量反馈噪声,并且能够近似模仿人工反馈,完善自适应学习机制的智能性。

关键词: 计算机应用;中文信息处理;自适应信息过滤;用户模型;相关反馈;阈值;层次聚类

中图分类号: TP391

文献标识码: A

Learning Algorithm of Adaptive Information Filtering Based on Hierarchy Clustering

HONG Yu, ZHANG Yu, LIU Ting, ZHENG Wei, GONG Cheng, LI Sheng

(Information Retrieval Lab, School of Computer Science and Technology,
Haerbin Institute of Technology, Haerbin, Heilongjiang 150001, China)

Abstract: This paper adopts an adaptive learning algorithm based on hierarchy clustering to update user profile, which continuously abstract the cancrroids of one class of optimum information from the feedback flow of system, which effectively shield the learning process from plenty of feedback noises produced by distorted threshold and sparseness of initial information, which also can imitate artificial feedback approximately to perfect the intelligence of adaptive learning mechanism.

Key words: computer application; Chinese information processing; adaptive information filtering; user profile; relevant feedback; threshold; hierarchy clustering

1 引言

随着搜索引擎技术的应用,人们找到了一条从海量信息中获取知识的捷径,但是伴随其产生的许多问题却不能仅仅依靠检索技术的改进得到很好的解决,其中最突出的两类问题是如何屏蔽垃圾信息和如何个性化推送信息,因此,更加智能化的信息过滤技术成为弥补这些缺陷的最佳助手^[1,2]。传统的信息过滤,如批过滤和信息路由,都需要大量初始信息训练用户模型,并且在处理信息的过程中欠缺自发的学习与更新能力,这就极大地制约了信息过滤技术在实际应用中的发展。智能性更强的自适应信息过滤技术是一项在初始信息相对稀缺的情况下高

效完成过滤任务并自动优化的课题,其通过在线学习反馈信息来更新用户模型,并时刻监控信息流与用户模型的相关度指标,同时从中选择相关度高于阈值的信息作为反馈。传统自适应信息过滤系统主要包含四个组成部分^[3,4],其相互关系如图 1 所示。

1) 用户模型: 描述用户需求信息的特征空间。用户模型的构造策略包括向量空间模型、浅层语义索引、n 元语法模型和树,其中向量空间模型是最常

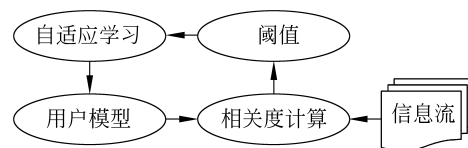


图 1 自适应信息过滤系统结构

收稿日期: 2006-08-24 定稿日期: 2006-12-20

基金项目: 国家自然科学基金资助项目(60435020,60575042,60503072)

作者简介: 洪宇(1978—),男,博士生,研究方向为自适应信息过滤,话题跟踪,个性化信息定制。

用的一种方法。

2) 相关度计算: 计算信息与用户模型的相似度。选择相关度计算的策略必须依据系统选择用户模型的方法, 比如: 向量空间模型的相似度一般采用特征向量空间的余弦夹角进行计算; 浅层语义索引则采用奇异值分解。

3) 阈值估计: 阈值是区分相关与不相关信息的边界, 阈值的存在显示信息过滤可以看作一种二元分类问题。

4) 自学习算法: 自学习机制的核心思想是通过反馈更新和改进当前用户模型的特征空间。目前效果较好的自学习策略包括 LR^[5] 和 Rocchio^[6,7] 等。

同比于人的学习习惯, 对于一个自适应过滤系统而言, 怎样获取最优信息进行学习是提高其学习效果的中心问题。传统的做法分为两类, 一类是完全依赖阈值的精准截取; 一类是凭借伪相关反馈的排序算法。基于这两种方法的学习机制在很大程度上提高了自适应信息过滤系统的智能, 但同时在实际应用中暴露了许多缺陷, 其中最为明显的两个问题是:

1) 阈值估计偏差问题^[8,9]: 早期的阈值估计一般都是在大规模语料中预先训练得到的, 这种阈值在过滤过程中不进行调整, 从而使判断信息相关性的过程存在偏见。为了应对自适应信息过滤的要求, 许多学者从事了阈值估计方面的研究, 比如 CMU 的 Yi Zhang^[9] 采用统计策略对阈值进行估计, 其观测到相关信息与用户模型的相关度成正态分布, 而不相关信息的相关度成指数分布, 并根据这种规律, 采用两种分布的联合概率估计阈值。该方法在 TREC 评测中得到的结果并不出色, 主要问题在于其不能考虑系统每次相关反馈对阈值的影响, 在用户模型时刻更新, 同时相关度指标整体浮动的环境下, 设置固定的阈值截取信息并不能有效解决偏差问题。此外, Yiming Yang 采用 ML R^[10] 算法, 在正例边界和反例边界之间的带状地带动态更新阈值。其问题在于两个边界逐渐归一并成递减趋势, 从而阈值的估计也恢复静态, 因此也不能彻底解决阈值偏差问题。

2) 伪相关反馈初始信息稀疏问题: 基于伪相关反馈的学习机制通常选择所有反馈, 或经过排序后相关性指标靠前的反馈更新用户模型。其缺陷在于忽视了用户模型先天的信息稀疏性。根据 TREC 对自适应信息过滤任务的定义, 每个用户模型的初始训练正例规模很小, 而在实际应用中, 用户通常也

不会给出需求信息的详细描述, 因此过滤结果的相关性指标并不能精确指向用户的真正意图。此外, 稀疏的初始信息赋予关键特征的上下文环境非常有限, 而语言本身又存在歧义性问题, 仅仅依靠统计学原理得到的相关性指标很有可能指向了一个错误的需求意图。基于这些因素, 传统的学习算法无法屏蔽反馈中大量的噪声并可能误导用户模型。

本文采用一种基于层次聚类的自适应学习机制, 通过对伪相关反馈进行聚类, 选择最优的一类信息更新用户模型, 从而削弱阈值估计偏见性和用户模型初始信息稀疏问题对过滤性能的影响。本文组织形式如下, 第二节介绍基于层次聚类的自适应信息过滤学习算法; 第三节介绍实验使用的语料及评价策略; 第四节介绍实验流程与安排; 第五节分析实验结果; 第六节结论。

2 基于层次聚类的自适应信息过滤学习算法

如第 1 节所论述, 制约自适应信息过滤学习机制效果的两个主要因素是阈值估计的偏差性和可供伪相关反馈对比的初始信息稀疏性。受这两个情况的影响, 伪相关反馈中相关度排序位置靠前的信息不一定满足真正的用户需求, 而相关度排序位置偏低的信息却有可能成为重要的相关信息。图 2 是采用 Rocchio 学习算法的自适应信息过滤系统针对用户模型的一次随机反馈记录, 横轴记录当前所有反馈信息与最优类质心的相关度, 纵轴记录所有反馈信息与用户模型的相关度。从图中我们可以观测到如下现象:

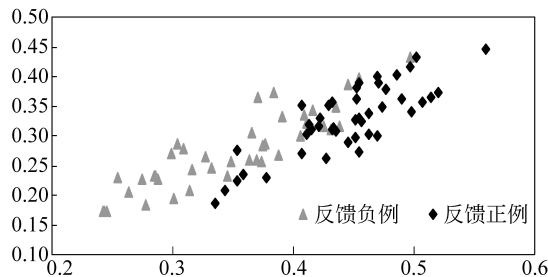


图 2 反馈分别与最优类和用户模型的相关度

正例: 经人工评测与用户模型相关的文本。

反例: 经人工评测与用户模型不相关的文本, 包括过滤系统判别为相关, 但人工评测不相关的文本。

最优类: 伪相关反馈经过聚类后与用户模型最相关的一类文本。实验采用向量空间模型描述各个聚类的质心, 并基于空间夹角的计算方法评价聚类与用户模型的相关性。

1) 当前反馈中,许多正例与用户模型相关度偏低,许多噪声与用户模型相关度很高。

2) 当前反馈中,正例与最优类质心相关度很高,噪声与最优类质心相关度偏低。

因此,如果选择所有伪相关反馈对用户模型进行更新,则会引入大量噪声信息;而截取相关度靠前的反馈更新用户模型,则遗漏了一定规模的正例信息。从另一个角度观察,一类最优的正例反馈与大部分噪声信息的相关性很低,同时包含了许多与用户模型相关度偏低的正例。这说明,上下文环境稀疏和特征歧义使用户模型与正例信息的匹配存在误差,这造成自学习模块无法正确抽取更有价值的反馈信息用于用户模型的更新,但反馈信息间内在的相关性却可以将正例反馈尽量聚集,同时疏远它们与噪声反馈的距离。因此,通过聚类获取一类最优反馈参与用户模型的学习,既可以屏蔽噪声反馈也可以减少学习中正例反馈的遗失。

本文陈述的基于层次聚类自适应信息过滤学习算法(以下简称 HCR)采用 BIRCH 算法对伪相关反馈聚类,选择最优的一类信息结合增量式 Rocchio 算法参与用户模型的学习。

2.1 增量式 Rocchio 学习算法

Rocchio^[6,7]学习算法利用相关信息的质心强化用户模型的正确特征,而利用不相关信息的质心削弱用户模型的噪声特征。该算法的定义如下式(1)。

$$\vec{p}(T) = \vec{q}(T) + \frac{\vec{d}_{D_+(T)}}{|D_+(T)|} - \frac{\vec{d}_{D_-(T)}}{|D_-(T)|} \quad (1)$$

其中,第一项是原始用户模型的特征向量空间;第二项是系统反馈的相关信息质心;第三项是相关度接近阈值的不相关信息质心。其中每一个向量都代表一篇文本的特征向量空间。公式中的 α 和 β

是控制用户模型、相关信息质心和不相关信息质心对学习过程影响强度大小的参数。

2.2 层次聚类算法

在聚类算法中,划分方法和层次方法^[11,12]是最常见的两类聚类技术,其中划分方法具有较高的执行效率,而层次聚类在算法上比较符合数据的特性。划分方法和层次方法的另外一个突出区别在于聚类前是否存在已知的类别信息:划分方法在聚类之前需要事先指定划分数 k 并确定初始划分;层次

聚类则不需要初始化,其在预先不知道目标集合内包含多少类别的情况下,自发地将所有信息聚合成不同的类。结合过滤过程分析,过滤系统提供的伪相关反馈包含多少类别是未知的,并且随着反馈信息的逐渐增加,类别的划分与数量也会随之变化,因此选择层次聚类方法嵌入过滤系统相对合理。常用的层次聚类算法包括 CHAMELEON 算法^[13]、CURE^[14]算法和 BIRCH^[15,16]算法等。HCR 学习模块选择 BIRCH 层次聚类算法嵌入自适应学习模块,其聚类步骤如下:

1) 将文档集 $D = \{d_1, \dots, d_i, \dots, d_n\}$ 中每篇文档 d_i 看作是一个具有单个成员的类 $c_i = \{d_i\}$,这些类构成了 D 的一个聚类 $C = \{c_1, \dots, c_i, \dots, c_n\}$;

2) 计算 C 中每个类对 (c_i, c_k) 之间的相似度 $\text{sim}(c_i, c_k)$;

3) 通过计算 $\arg\max \text{sim}(c_i, c_k)$ 选取相似度最大的类对 (c_i, c_k) 合并为新类 $c_0 = c_i \cup c_k$,从而构成 D 的一个新的聚类 $C = \{c_1, \dots, c_n\}$;

4) 返回步骤(2),直到 C 中只剩下一个类或达到一个指定条件为止;

5) 返回层次聚类结果。

2.3 结合层次聚类的 Rocchio 学习算法

采用增量式 Rocchio 学习机制的传统自适应过滤算法中,陆续进入过滤系统的信息首先与用户模型进行相关度匹配,如果相关度大于先验阈值,则作为正例输入过滤系统的自学习模块并反馈给用户,否则作为不相关信息被过滤掉。实际上,系统需要实时地保留若干不相关信息作为 Rocchio 学习中的反例。过滤系统可以预先设置经验性规模的文本池,将被判定为不相关的信息实时地嵌入该文本池,并对其中所有文本进行相关度排序,选择相关度最高的不相关信息(相比于相关度很低的信息,与用户模型很相似的不相关信息中包含更多的干扰性特征,因此作为反例输入自学习模块更有利于用户模型的调整)作为反例输入 Rocchio 学习模块,而相关度最低的文本将从该文本池中删除。过滤过程中每次检测到正例信息时,系统都对记录下来的相关信息和不相关信息分别计算质心,并利用公式(1)对用户模型进行调整。在这个过程中,随着正例数量逐渐增加,用户模型特征空间中与需求相关的特征成增量式变化,而反例信息则抑制和削弱特征空间中的噪声信息。

HCR 在 Rocchio 算法的基础上嵌入 BIRCH 聚

类算法,其组成如图 3 所示。当过滤系统向 HCR 发送一则伪相关反馈时,BIRCH 首先对当前接收到的所有伪相关反馈进行聚类;聚类择优部分对 BIRCH 得到的每个聚类计算质心并评价每个类与用户模型的相关度;Rocchio 模块将最优类、用户模型原型以及反例质心作为输入,采用公式(1)获得更新后的用户模型。

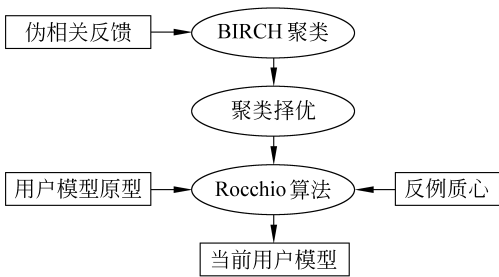


图 3 基于 BIRCH 层次聚类的 Rocchio 学习算法 HCR

HCR 并不是利用聚类获取相关反馈的算法,在学习过程中既不更改过滤系统已经获得的伪相关反馈现状,也不利用信息流与聚类的相关度裁决信息的取舍,其核心作用是利用信息内在的上下文环境,从当前的伪相关反馈中抽取更优的一类信息进行学习并更新用户模型。

HCR 设置临时用户模型(以下简称 T-Profile)和用户原型(以下简称 O-Profile),前者保存每次自学习的结果 $P(i)$,后者则持续保存用户需求的初始模型。每次自学习的过程中,HCR 都将恒定的 O-Profile 输入 Rocchio 模块,而将更新得到的用户模型 $P(i)$ 输入 T-Profile, $P(i)$ 在 T-Profile 中一直保存到下一次系统反馈,此时 T-Profile 被更新成 HCR 最新产生的用户模型 $P(j)$ 。采用这种更新方法的原因在于最优类的漂移性,由于伪相关反馈可能包含多个可以用来优化用户模型的候选聚类,而这些聚类与用户模型的相关强度随着反馈信息的不断增加将会漂移,曾经相关强度次优的聚类可能在下次反馈结束后获得最优的候选价值。采用这种更新方法的 HCR 可以回避持续迭代更新用户模型造成的偏见性,同时削弱信息流输入顺序引起的信息遗漏。

3 语料及评测

3.1 语料

实验选择天网 100 G 作为训练和测试语料,同时使用 2005 年参加 863 评测的检索系统(以下简称

IR863-System)针对 30 个主题分别进行检索。检索结果由 10 名学生进行评测,其中每两名学生为一组同时对 6 个主题进行人工评测。此后,试验投入 6 名学生对评测结果进行校验,得到关于 29 个主题的 3 937 篇正例文档,其中一个主题没有发现相关文档。

从原始语料中,我们随机抽取了 17 840 篇文档,并与人工评测得到的相关文档融合。此后,试验分别选择 25 % 加入训练与开发集,50 % 加入测试集,以下将该语料简称为 Emur-Corpus。该语料的特点是具有与真实语料相似的时空顺序,适于检测 HCR 是否能够在接近真实环境中保证过滤性能。此外,实验还构造了一个规模略小的语料。首先使用 IR863-System 在全集语料中对每个主题进行检索,其反馈的结果按照相关度进行排序。然后,选择每个主题的前 500 篇文档与人工评测结果取并集。最后,选择同上的比例关系将语料划分为训练、开发和测试语料集。该语料的特点是其包含了大量相关度指标非常接近甚至超过正例文档的反例信息。该特点非常适合检测 HCR 屏蔽反馈信息噪声和削弱过滤阈值偏差的性能。以下将该语料简称为 Rak-Corpus。

3.2 评测

该实验采用的评测方法是 TREC-11 metrics,简称为 T11SU。T11SU 在 TREC 评测以及 TDT 的研究与实验中被广泛采纳,该方法的相关定义及流程如下:首先,实验将信息过滤的结果可以分为四种情况,如表 1 所示。其中 R^+ 代表系统从信息流中筛选出相关文本的数量; N^+ 代表系统筛选出不相关文本的数量,即错检指标; R^- 代表被系统过滤掉但相关的文本数量,即漏检指标; N^- 代表被系统过滤掉且不相关的文本数量;其次将 A、B、C、D 定义为上述四种指标对性能的影响强度。则信息过滤任务的评测公式定义如下:

$$Utility = A \times R^+ + B \times N^+ + C \times R^- + D \times N^-$$

(2)

表 1 信息过滤结果分类

	实际相关	实际不相关
系统认为相关	R^+ / A	N^+ / B
系统认为不相关	R^- / C	N^- / D

而在 TREC-10 和 TREC-11 中,评测公式被明

确定义为：

$$T11SU_{\text{}} = \frac{\max \left(\frac{R^+ - \alpha N^+}{R^+ + R^-}, 0 \right)}{1 - \alpha} \quad (3)$$

其中：参数 α 控制评测系统对错检指标影响过滤性能的重视程度，参数 α 用以平滑评测指标。TREC-10 和 TREC-11 将 α 分别设置为 0.5 和 -0.5。

4 实验安排与流程

本节分别介绍 HCR 训练流程及测试流程。训练阶段，实验在训练语料上估计用户模型的向量空间维度 D 、反例列表容量 N 、Rocchio 参数以及层次聚类阈值；在测试阶段，实验首先在开发语料上调整和优化过滤系统的阈值，然后在测试语料上分别评测 Rocchio 和 HCR 的性能。

4.1 训练流程

HCR 的训练过程需要估计三组参数：Rocchio 参数、空间规模和聚类阈值。

1) Rocchio 参数采用最小均值平方 (Least Mean Squares) 策略进行训练。首先将公式 (1) 表示成线性函数 $h(x)$ ：

$$h(x) = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}^T \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} \quad (4)$$

其中， $\begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix}$ ； x 各项依次代表主题对应的初始模型、训练语料正例质心和反例质心。其次，定义差异函数 (Cost Function) $J(\theta)$ ：

$$J(\theta) = \frac{1}{2} \sum_{i=1}^m (h(x^{(i)}) - y^{(i)})^2 \quad (5)$$

其中， $h(x^{(i)})$ 代表主题 i 的线性函数 $h(x)$ ， $y^{(i)}$ 代表主题 i 的 Profile 原型。最后，根据梯度衰减原理 (Gradient Descent Algorithm) 的定义，对于每个参数采用如下更新函数：

$$\theta_j = \theta_j - \frac{\partial}{\partial \theta_j} J(\theta) \quad (6)$$

系统对参数初始化之后，根据公式 (6) 循环迭代参数直到收敛。训练过程中，系统循环将每个主题的 x 代入公式 (6)，每对一个主题训练结束后，当前的指标将作为针对下一个主题训练的初始值。

2) 空间规模参数涉及特征空间维度 (S-num) 和反例列表容量 (N-num)，在训练 Rocchio 参数的过程中，这两个指标初始化为相对较高的值。在随

后的训练中，系统以适当颗粒度逐渐递减它们的值，同时使用 T11SU 在每个点上评测过滤系统的性能，最后选择性能最优的情况设置它们的指标。

3) 聚类阈值训练中，系统将人工评测的 29 个主题作为评价聚类效果的标准，同时对融合了 3 937 篇正例文档的 Emu-Corpus 和 Rak-Corpus 分别训练。训练过程中，系统将 29 个主题看作 29 个已知类，对聚类得到的所有类别计算质心，并针对每个已知类从中选择质心最相近的聚类结果进行对比，计算其 F 测度值，最后将所有 29 个已知类的 F 测度值求和取平均，从而得到当前聚类阈值对应的聚类性能。在该试验中，训练系统将聚类阈值初始化为 0.000 5，并以 0.000 1 的颗粒度递增，每变化一次，计算一次聚类性能，最后以最佳聚类指标对应的阈值作为训练结果。

Rocchio 参数及空间规模参数的具体指标请参考表 2，阈值参数请参考第五节中的表 3。

表 2 Rocchio 参数及空间规模参数训练结果

语料				S-num	N-num
Emu-Corpus	0.5	0.3	0.2	2 000	50
Rak-Corpus	0.4	0.3	0.3	2 100	45

4.2 测试流程

实验分别对 Emu-Corpus 和 Rak-Corpus 进行两组测试：

1) 在开发集中，以 0.01 为颗粒度逐渐微调过滤阈值，并记录 Rocchio 和 HCR 对应每个阈值的过滤性能。

2) 在测试集中，应用训练阶段得到的相关参数并选择开发集中最优的过滤阈值对更大规模的测试语料进行测试，分别评测 Rocchio 自学习机制和 HCR 学习机制的性能。

该实验主要检测 HCR 是否能够在性能上超越传统 Rocchio 算法的性能，从而验证基于层次聚类的自适应学习能够在初始用户模型上下文稀疏的情况下，尽可能多地获取实际相关的一类信息对过滤模型进行改进，同时屏蔽阈值估计偏差造成的大量反馈噪声。

5 实验与分析

第一组实验使用 Emu-Corpus 开发集分别对

Rocchio 和 HCR 学习算法进行测试。过滤系统在开发集上以 0.01 的颗粒度调整过滤阈值,并记录每个阈值点上两个算法的性能。实验结果如图 4 所示,其横轴记录过滤阈值,纵轴记录 T11SU 评测指标。

从整体上观察,基于层次聚类的 Rocchio 学习机制并没有在性能上取得大幅度的提高,其原因正如 3.1 节论述的情况,由于 Emur-Corpus 是由正例文档和随机从原始语料中抽取的文档组成,同时规模远远小于语料全集,因此相关性指标接近正例文档集的反例很少甚至没有被选进开发和测试语料。在这种正例和反例相关性指标差距本身很大的语料中,只要恰当设定过滤阈值,基于增量式 Rocchio 的过滤系统就可以达到很高的过滤性能。但从图 4 中,我们观测到两种现象:一个是 HCR 过滤系统的性能在每个测试点上都不低于增量式过滤系统,这说明一类小规模最优的反馈可以替代所有伪相关反馈对用户模型进行更新;另一个现象是在阈值指标相对较低的部分,HCR 过滤系统性能大幅超越没有嵌入聚类学习的 Rocchio 过滤系统,最大时达到 25.9 个百分点。这说明当过滤阈值估计不精确的时候,由于阈值偏见性引起的大量噪声通过嵌入聚类择优策略有效地被屏蔽掉了。

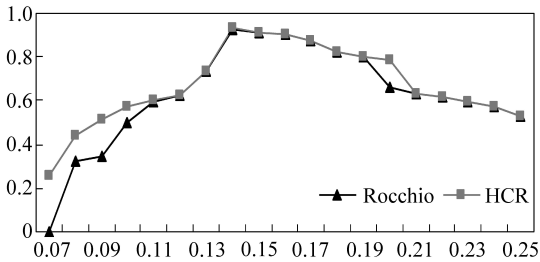


图 4 HCR 与 Rocchio 基于 Emur-Corpus 的性能对比

第二组实验使用 Rak-Corpus 开发集分别对 Rocchio 学习算法和 HCR 学习算法进行测试,试验结果如图 5 所示,其横轴记录过滤阈值,纵轴记录 T11SU 评测指标。

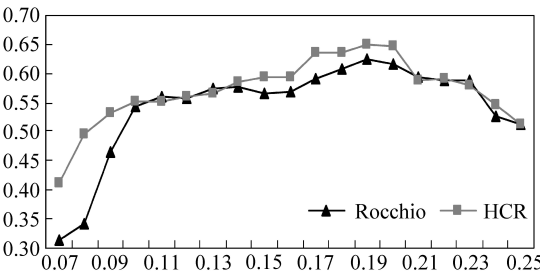


图 5 HCR 与 Rocchio 基于 Rak-Corpus 的性能对比

这组试验结果与第一组试验相比,最明显的差别在于最优过滤阈值附近的区间内,HCR 系统的性能明显高于 Rocchio 系统。正如 3.1 节所述,Rak-Corpus 中大量反例与用户模型的相关性指标很高,有相当一部分反例文档的指标近似于甚至超越了正例文档,在这种情况下,采用统计策略裁决反馈信息对更新用户模型的价值将存在偏差。HCR 利用反馈文档自身的上下文环境和相互之间的关系抽取一类最优文档更新用户模型,从而将反馈中大量与用户模型相关性统计指标很高,但与最优类相关性偏低的反例文档屏蔽掉;同时反馈中与用户模型相关性统计指标偏低,但与最优类相关性很高的正例文档可以被抽取并用于更新用户模型。因此,虽然 Rocchio 过滤系统的性能已接近自身的最优,但嵌入聚类择优模块后,整体性能又取得了大幅提高。

根据 Rocchio 和 HCR 在开发集上的测试,分别对它们在 Emur-Corpus 和 Rak-Corpus 测试集上的试验设置参数,测试结果如表 3 所示。其中,F-threshold 代表过滤阈值;C-threshold 代表聚类阈值;Emu 代表 Emur-Corpus 的测试语料,Rak 代表 Rak-Corpus 的测试语料。

表 3 基于 Emur-Corpus 和 Rak-Corpus 测试集的实验结果

算法 3 语料	F-threshold	C-threshold	评测结果
Rocchio/ Emu	0.14	—	0.915 9
HCR/ Emu	0.14	0.007 5	0.918 4
Rocchio/ Rak	0.19	—	0.610 4
HCR/ Rak	0.19	0.015 9	0.639 6

如表所示,嵌入聚类择优算法的 Rocchio 学习机制 HCR 在 Emur-Corpus 的测试集上为过滤系统带来近似 0.3 个百分点的提高;在 Rak-Corpus 的测试集上获得近似 3 个百分点的提高。此外可以观测到,在 Rak-Corpus 开发集上的最优过滤阈值点上,HCR 过滤系统提高了近似 2.5 个百分点,比 HCR 在规模稍大的 Rak-Corpus 测试集上获得的性能提高低了 0.5 个百分点,这说明聚类择优更新机制的优化效果非常明显。同时我们也发现 Rocchio 和 HCR 在不同语料集上评测指标差异很大,这说明初始信息稀疏和过滤阈值偏差对过滤性能的影响非常剧烈,如何更加有效地解决这一问题仍将是自适应信息过滤研究中非常重要的课题。

传统基于 Rocchio 的自适应学习模块需要保存所有伪相关反馈。在此基础上嵌入聚类算法的学习

模块并不保存历次聚类的结果,而是在过滤系统检测到一篇相关文本后实时进行再聚类,因此 HCR 系统并没有增加额外的空间开销。但是每次更新用户模型时,学习模块都需要花费额外的 $O(n \log n)$ 时间复杂度进行聚类,因此在一定程度上影响了过滤系统的处理效率。但是过滤系统的一个优点在于伪相关反馈规模很小(实验中的规模介于 3 ~ 288 篇),因此 HCR 系统的时间消耗并不影响实际应用中的用户需求(该系统已经应用于本研究室于 2006 年 5 月正式发布的人网系统: <http://pic.ir-lab.org/pic/index.html>,对于每天 crawler 爬行得到的最新信息流可以在 30 分钟内完成过滤处理,并向参与定制信息的注册用户推送相关网页)。

6 结论

本文介绍了一种嵌入层次聚类的自适应学习算法,通过聚类择优模块选择一类最优的反馈信息参与用户模型的学习,从而有效改进过滤系统的学习性能。经过实验验证,该方法一定程度上屏蔽了初始用户模型上下文稀疏以及语言歧义问题对自适应学习的误导。此外,选择聚类择优后的一类信息参与 Rocchio 自适应学习与更新,能够有效削弱阈值和伪相关反馈排序偏差造成的负面影响。

参考文献:

- [1] 王斌,潘文锋. 基于内容的垃圾邮件过滤技术综述[J]. 中文信息学报,2005(5): 1-10.
- [2] Belkin NJ, Croft W B. Information filtering and information retrieval: two sides of the same coin[J]. Communications of ACM, 1994, 35(12): 29-38.
- [3] Robertson S and Soboroff I. The TREC-10 filtering track final report [A]. In: Proceeding of Tenth Text Retrieval Conference [C]. Gaithersburg, USA: MD, 2001, 26-37.
- [4] Robertson S and Soboroff I. The TREC-11 filtering track final report [A]. In: Proceeding of Eleventh Text Retrieval Conference [C]. Gaithersburg, USA: MD, 2002, 26-37.
- [5] Yang Y, Yoo S, Zhang J, Kisiel B. Robustness of Adaptive Filtering Methods In a Cross-benchmark Evaluation [A]. In: Proceedings of the 28th annual international ACM SIGIR [C]. Salvador, Brazil: ACM Press, 2005, 33-39.
- [6] Ault T, Yang Y. Knn, rocchio and metrics for information filtering at trec-10[A]. In Proceeding of Tenth Text Retrieval Conference [C]. Gaithersburg, USA: MD, 2001, 84-92.
- [7] Allan J. Incremental relevance feedback for information filtering [A]. In: Proceedings of the 19th annual international ACM SIGIR [C]. Zurich Switzerland: Center for Intelligent Information Retrieval, 1996, 270-277.
- [8] Zhang Y and Callan J. The bias problem and language models in adaptive filtering [A]. In Proceeding of Tenth Text Retrieval Conference [C]. Gaithersburg, USA: MD, 2001, 34-41.
- [9] Zhang Y and Callan J. Maximum likelihood estimation for filtering thresholds [A]. In The Twenty Fourth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02) [C]. New York: The Association for Computing Machinery. 2002, 294-302.
- [10] Yang Y, Kisiel B. Margin based Local Regression for Adaptive Filtering [A]. In: Proceedings of the twelfth international conference on Information and knowledge management [C]. New Orleans, Louisiana, USA: CIKM, 2003, 88-95.
- [11] 甄彤. 基于层次与划分方法的聚类算法研究[J]. 计算机工程与应用, 2004, 01(6): 178-180.
- [12] 苏中, 马少平, 杨强等. 基于 Web-Log Mining 的 Web 文档聚类[J]. 软件学报, 2002, 13(01): 99-104.
- [13] 吴帆, 李石君. 一种高效的层次聚类分析算法[J]. 计算机工程, 2004, 30(9): 70-71.
- [14] Wang L, Kitsuregawa M. Use Link-based Clustering to Improve Web Search Results [A]. In: Proceedings of the second International Conference on Web information Systems Engineering [C]. Washington, DC: WISE, 2001, 119-128.
- [15] Zhang T, Ramakrishnan R, Livny M. BIRCH: an efficient data clustering method for very large databases [A]. In: Proceedings of the 1996 ACM SIGMOD international conference [C]. Montreal: ACM Press, 1996, 103-114.
- [16] Zhang T, Ramakrishnan R, Livny M. Zhang. BIRCH: a new data clustering algorithm and its applications [J]. Journal of Data Mining and Knowledge Discovery, 1997, 1(2): 141-182.