第33卷 第14期 计 算 机 工 程 2007年7月
Vol.33 No.14 Computer Engineering July 2007
·人工智能及识别技术· 文章编号：1000—3428(2007)14—0177—03 文献标识码：A 中图分类号：TP391.9

# 一种基于 Agent 的文本情报检索模型

谭天晓，赵 辉，赵宗涛
(西安高技术研究所计算机室，西安 710025)

**摘 要：**信息战中，情报是决定战争成败的关键点之一。针对文本情报，构建了一个基于多 Agent 的检索模型 AIRSM，并结合潜在语义技术，进行用户建模，以满足不同用户的检索需求。实验证明，AIRSM 在一定程度上实现了军事情报检索的个性化、智能化。
**关键词：**Agent；AIRSM；情报检索

# Text Intelligence Retrieval Model Based on Agent

TAN Tianxiao, ZHAO Hui, ZHAO Zongtao
(Division of Computer Science, Xi'an Research Institute of Hi-tech, Xi'an 710025)

**【Abstract】** Intelligence is one of deciding factor in informational combat. This paper presents a retrieval model——AIRSM based on multi-agent, contemporarily, it constructs a user model through the method of latent semantic indexing(LSI), AIRSM can adapt to retrieval demands of different users. The experiment proves that AIRSM can achieve military intelligence retrieval intelligently and individually to a certain extent.
**【Key words】** agent; AIRSM; intelligence retrieval

近年来，随着军事情报侦察的信息化程度不断提高，各情报侦察单元在长期的情报信息收集、管理过程中，建立了大量的数据库用于保存原始数据和处理结果。如何在庞杂的信息资源中快速、准确地找到所需要的情报，对于各级指挥决策人员至关重要。现有的军网搜索引擎普遍存在精度不高、不能提供个性化服务等问题。本文提出一种基于 Agent 的文本情报检索模型 AIRSM，将智能体技术与用户模型有机地结合，为解决上述问题提供了新的思路。

## 1 AIRSM 的总体设计

### 1.1 AIRSM 的体系结构

情报检索系统处于用户和联指情报中心网络资源之间，由用户接口 Agent、情报处理 Agent、情报收集 Agent 和用户模型 Agent 4 个模块组成。逻辑上用户接口 Agent 处于最上层。

信息处理 Agent 和信息收集 Agent 处于中间层；用户模型 Agent 处于最底层。共同完成用户兴趣学习、信息收集、信息过滤以及信息显示与反馈等功能。各个模块之间没有互相交叠的部分，也没有共享的公共数据区，只通过通信机制完成模块之间的相互调用，较好地做到了模块之间的松散耦合，有利于软件的开发、测试、修改和重用等。其体系结构如图 1 所示。

各 Agent 的工作流程为：

(1)用户接口 Agent 获得用户提交的查询关键词或语句，并提交给情报处理 Agent；

(2)情报处理 Agent 对查询语句进行处理，从本地信息库查找出相关的文档给用户，或将处理结果传送给情报搜集 Agent；

(3)情报收集 Agent 带着处理过的查询请求借助军用搜索引擎进行相关搜索，获取相关网页的链接存放在 URL 队列缓冲区；

(4)情报搜集 Agent 访问 URL 队列缓冲区，取出链接地址将相关文本信息下载至本地；

(5)用户模型 Agent 根据其他 Agent 提供的数据负责建立、维护、更新用户模型；

(6)情报处理 Agent 参照用户兴趣模型进行文档匹配过滤，将过滤结果按照兴趣主题进行分类、索引并存储到本地 Web 信息库，并在需要的时候提交给用户接口 Agent，同时完成自主学习，信息推荐及决策；

(7)用户接口 Agent 将查找到的信息提供给用户，收集用户可能提交的反馈信息；

(8)用户模型 Agent 根据用户的反馈信息更新、完善用户的兴趣模型。

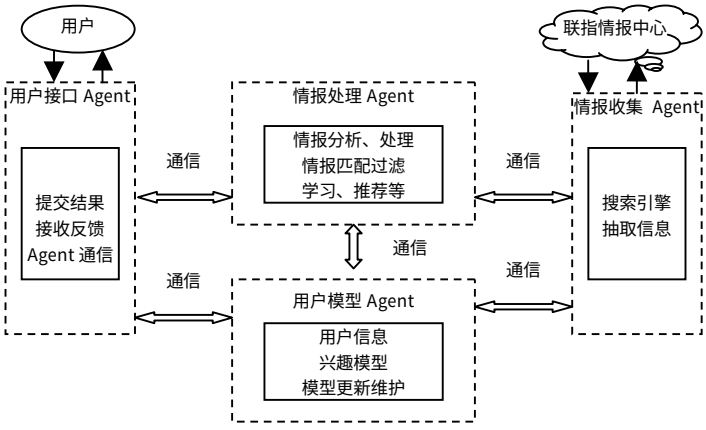


**图 1 AIRSM 的体系结构**

### 1.2 AIRSM 各模块功能

(1)用户接口 Agent。用户接口 Agent 主要实现用户与检

**作者简介：**谭天晓(1975－)，女，博士、工程师，主研方向：人工智能，信息系统建模；赵 辉，博士、工程师；赵宗涛，教授、博士生导师
**收稿日期：**2006-07-25 **E-mail：**ttx_erpao@sina.com

索系统的交互，将用户的查询请求提交给系统以及将系统检索到的信息呈现给用户。接口 Agent 具有反馈机制，由用户对所检索的信息进行满意程度的评价。接口 Agent 同时负责与情报处理 Agent 和用户模型 Agent 进行通信以交换、传递必要的信息。

(2)情报处理 Agent。信息处理 Agent 是 AIRSM 的“ 任务执行者 ”，主要是对系统中的信息进行分析处理：

1)对情报文本或者 Web 情报文本进行分析处理，如分词、停用词的处理，然后按照系统的设定对文本进行特征项和特征项权重值的计算，将文本表示为向量格式，方便计算机处理；

2)分析信息收集 Agent 搜索到的信息，参考用户的个性化兴趣模型过滤掉不相关的或用户不感兴趣的文档，将符合用户需求的信息呈现给用户；3)进行自主学习、信息推荐决策。

(3)用户模型 Agent。用户模型 Agent 主要由用户个人信息库、特征信息库、规则库、兴趣模型文档库等组成，负责建立、维护、更新用户模型。并在其他 Agent 需要使用用户模型信息时，提供给它们。

(4)情报收集 Agent。情报搜集 Agent 主要负责从联指情报中心网络上搜索用户需要的信息。为了尽量提高文本情报的查全率，可在将查询语句转换成多个搜索引擎的查询语句参数的同时进行多个进程的检索。同时为了实现并发查询，可为每个代理引擎建立一个线程，每个线程分别代表一个搜索引擎的查询，由页面分析程序提取出网页的链接记录到一个临时队列中，在对重复、冗余的链接处理后，由情报收集 Agent 访问这些超链接所指向的 Web 文档，并将文档的标题和正文交给情报处理 Agent 进行处理并保存到信息库中。

### 1.3 AIRSM 通信机制

在 AIRSM 中，4 个 Agent 是独立工作的，当它们需要传递消息和合作完成任务时，要进行 Agent 间的通信。本文按照消息/对话模式来设计 Agent 的通信机制，消息传递和处理机制如图 2 所示。

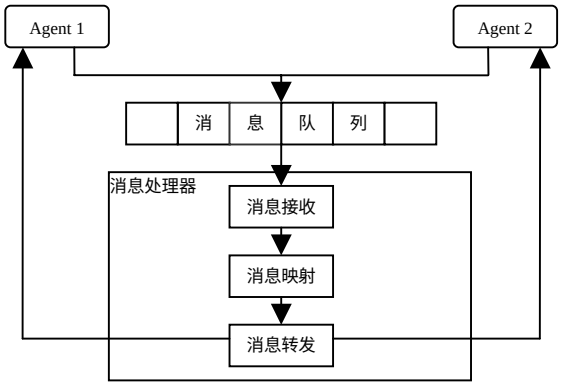


**图 2 AIRSM 通信机制**

从图 2 中可以看出，消息处理器负责消息的接收、映射、转发等工作。当一个 Agent 需要获得其他 Agent 的合作时，将发送一条消息，这条消息暂存到系统的消息队列中。消息处理器在消息队列为空时，处于等待状态，一旦消息队列被填入消息，它立刻被唤醒，依次处理队列中的消息，直到处理完毕消息处理器再一次进入等待状态。消息处理器接收到消息后，对消息进行分析，本文设计了一个“ 消息映射表 ”，表中指定了消息的类型、消息编号、能处理该消息的 Agent 的信息等，见表 2。每个 Agent 启动时，要向“ 消息映射表 ”填入自己的信息、能处理消息的类型、编号等(即进行注册)。消息处理器根据“ 消息映射表 ”实现消息的路由、转发。

**表 1 消息映射表示例**

| 消息编号 | 消息类型 | Agent 名 | Agent 地址 | … |
|---|---|---|---|---|
| 1860 | A | “Agent1” | &Agent1 | … |
| 1861 | B | “Agent2” | &Agent2 | … |
| … | … | … | … | … |

在这里，每个 Agent 不直接与其他 Agent 交换消息，只与消息处理器发生联系。消息处理器知道谁能完成什么任务，应把消息转发给谁。这种方式节省了系统处理的时间和空间，提高了处理效率。另外，“ 消息映射表 ”提供了一个开发的结尾，从而可以活动地增加或减少这个映射表的内容，当 Agent 需要增添或删除某些功能时，只需要改变 Agent 自身，系统其他部分不会受到影响，这种方法很好地符合了软件工程的思想。

## 2 AIRSM 用户建模

AIRSM 要实现高度的智能化、个性化，还需要高效率的用户兴趣模型。系统利用用户模型中包含的知识来剪裁它的界面以适合特定用户的需求，使系统在检索过程中能根据相应的用户模型，提出合适的意见和检索策略。AIRSM 是通过用户给出的示例文本和浏览并保存的 Web 页面来获取不同用户个性化信息的。

### 2.1 基于 LSI 的用户模型表示

通过对现有用户模式构建的研究，结合潜在语义索引技术，本文设计了一种基于潜在语义的用户兴趣模式构建机制。潜在语义索引(LSI)是一种概念检索方法[1]，它通过奇异值分解计算，将索引项、文档和检索表达式按照语义相关程度组织在同一个语义空间中，在这一语义空间中，分散在不同文档和检索表达式中的同义词之间的距离相近，主题语义接近的文档和检索表达式位置相邻。索引项、文档和检索表达式之间的联系就是它们之间的潜在语义。

基于 LSI 的用户兴趣模型最初由用户提供的示例文本集的特征词频矩阵构成。其结构示意图见图 3。

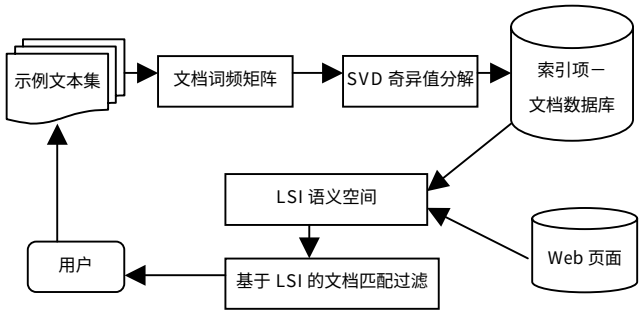


**图 3 基于 LSI 的用户模型**

在本模型中，首先由用户提供相应兴趣主题的示例文本集，通过对样本文档进行分词、消除停用词处理后，生成每篇文档中的特征项集，将一个兴趣主题类别中所有文档的特征项统一为原始特征空间，计算出每个特征项表达该兴趣主题的权重值，并按权重值大小排序，按设定的阈值取适当的特征项数作为用户在该兴趣主题的信息表示。因此，用户模型首先表示为一个文档词频矩阵，这部分的算法为：

输入　每个兴趣主题的样本文档$D_j$和设定的特征项个数 num；

输出　能够反映用户兴趣的特征词库和词频矩阵；

Step1 训练文本集中依次取得每个文本，调用分词程序，将其分词，并去除停用词。保留名词和动词，因为只有名词和动词才有可能是特征项；

Step2 调用特征提取算法，提取出文档特征项；

Step3 计算特征项的权重值，按照设定的 num 值取相应特征项数构建特征词库；

Step4 根据特征词库，为每篇文档生成一个映射(关键码，值)。关键码为特征词，值为该特征项在文本集中的权重值；

Step5 生成每个文本的特征向量，构建出文档－词频矩阵。

生成的词频矩阵 $\boldsymbol{A}$(m*n)中，m 表示文档集中包含的所有不同特征词的个数；n 表示该类文档集中的文档数。每一个特征词对应于矩阵 $\boldsymbol{A}$ 的一行；而每一个文档则对应于矩阵 $\boldsymbol{A}$ 的列。此时矩阵中的值已不是简单的特征项出现次数，而是经过权重算法得到的特征项的权重值。词频矩阵 $\boldsymbol{A}$ 建立后，按照矩阵论中的奇异值分解技术(SVD)，$\boldsymbol{A}$ 可分解为

$$\boldsymbol{A}=\boldsymbol{TSD}^{\mathrm{T}} \tag{1}$$

式中，$\boldsymbol{T}$、$\boldsymbol{D}$ 各列正交且长度为 1，即 $\boldsymbol{TT}^{\mathrm{T}}=1, \boldsymbol{DD}^{\mathrm{T}}=1$；$\boldsymbol{S}$ 是奇异值的对角矩阵，$\boldsymbol{S}=diag(\lambda_1,\lambda_2,.....\lambda_t)$；$\lambda$ 为对角矩阵的特征值。

选取 $\boldsymbol{S}$ 前 K 个最大的奇异值，其余设为 0，同时删除 $\boldsymbol{T}$ 和 $\boldsymbol{D}$ 中相应的列，可以得到秩为 K 的新矩阵。

$$\boldsymbol{A}_K=\boldsymbol{T}_0\boldsymbol{S}_0\boldsymbol{D}_0^T \tag{2}$$

这一部分的具体算法如下：

输入　词频矩阵 $\boldsymbol{A}$，设定的 K 值；

输出　近似矩阵$\boldsymbol{A}_{\mathrm{k}}$以及$\boldsymbol{T}_0$, $\boldsymbol{D}_0$和$\boldsymbol{S}_0$；

Step1 输入 A，调用奇异值分解程序，得到词频矩阵的左右奇异向量矩阵和对角阵；

Step2 根据设定的k值，分别取左右奇异矩阵和对角阵的前k列，利用公式产生k秩近似矩阵$\boldsymbol{A}_{\mathrm{k}}$；

Step3 输出基于LSI的索引矩阵$\boldsymbol{A}_K$以及$\boldsymbol{T}_0$, $\boldsymbol{D}_0$和对角　阵$\boldsymbol{S}_0$。

在这个算法中，关键是 K 值的确定。考虑到向量运算的响应速度和存储空间的限制，可以对空间维数的数值规定上限，实际要通过多次试验，才能确定针对具体文档集合操作效率好的 K 值。

利用 LSI 方法得到关于用户兴趣的索引－文档集合后，将与查询匹配的 Web 页面映射到 LSI 语义空间中，计算兴趣主题文档集向量与新文档向量之间的相似值，如果该值大于设定的阈值则文档是用户所需要的；反之，则是用户不感兴趣的。与通常的用关键词描述的兴趣主题信息不同，在 LSI 方法中兴趣主题模型是通过降维后的词频文档矩阵来表示的，通过奇异值分解得到的 k 个正交因子在一定程度上隐含了该兴趣主题的语义信息。

### 2.2 用户模型更新

对一个已经存在的LSI用户模型，如果需要加入新的文档和索引词，最直接的办法是重新建立词频矩阵然后进行SVD计算。但是SVD分解的计算量非常大，重新进行SVD计算需要更多的时间，所以在实际应用中，LSI用户模型的更新一般采用folding-in算法来实现[2]，folding-in算法能在已经存在的潜在语义空间中加入新的文档和索引词而不影响现有文档和索引词的结构，该算法要求在加入新的文档和索引特征项前须对这些文档和特征项进行预处理，将其转换成k维空间中的向量。当然如果新加入的文本和索引特征项过多时也应重新进行SVD计算，重新构建新的语义空间。

### 2.3 相关反馈

系统只有在与用户的不断交互中获取用户的反馈信息，才能进一步获取用户的兴趣信息，及时对用户兴趣模型修改完善。在系统原型设计中，为交互界面设置一个评分模块：其取值范围设为[0,1]，用户在浏览完系统推荐的或过滤后的文档信息后，可以通过评分模块来对页面进行评价，满意程度用 $\lambda$ 表示。$\lambda=1$ 表明用户对该页面非常感兴趣；$\lambda=0$ 表示用户对此页面不感兴趣。$\lambda$ 值越高，用户对该页面越感兴趣；$\lambda$ 值越低，表明用户对此页面越不感兴趣。当反馈的相关文本达到一定的数量时，可以添加到用户兴趣模型的文档库中，重新进行计算并构建出新的兴趣模型以适应用户兴趣的变化，改进系统过滤的效果。

## 3 实验结果

AIRSM 原型是在 Windows 系统下，采用 Visual C++6.0 开发实现的。为了验证系统处理的性能，本文利用了联指情报中心 2004 年 5 月的一部分历史数据，包括军事理论、武器装备、边海空情以及热点冲突等情报文档，作为实验数据。实验过程中，原型系统能够良好地执行情报查询匹配过滤、用户模型的构造和维护、Agent 机器学习及通信等任务。图 4 为统计 10 类情报数据 300 多份文档得出的 AIRSM 查全率与查准率比较。

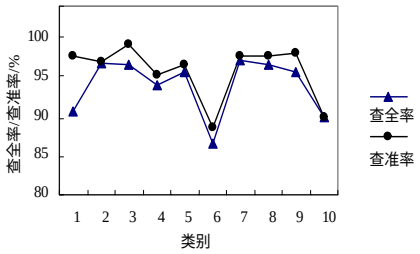


**图 4　AIRSM实验结果**

从图中看出，AIRSM 检索相似度好，查全率与查准率两个评测指标也较高，结果还是比较令人满意的，但系统在某些算法上还存在需要改进的地方。

## 4 结束语

将 AIRSM 应用于军事情报检索，系统不仅具有高度的自治性、反应性、自发性及适应性，能够在没有用户干预的情况下，主动搜集用户信息，而且能对用户的信息需求进行分析，帮助用户表达个性化需求。这充分体现了 Agent 技术与用户建模技术相结合的优越性。由于条件所限，本文只针对文本情报的检索进行了研究，随着军事信息一体化的发展，还有许多方面值得进一步探讨，如智能体通信、人机交互、机器学习以及多媒体情报检索。

## 参考文献

1 朱巧明, 李培峰, 吴　娴. 中文信息处理技术教程[M]. 北京: 清华大学出版社, 2005.

2 Berry M W, Dumais S T, O'Brien G W. Using Linear Algebra for Intelligent Information Retrieval[J]. SIAM Review, 1995, 37(4): 573-595.

3 刘飞飞, 刘军万. 基于潜在语义索引的文本结构分析方法的研究[J]. 情报杂志, 2004, 23(1): 56-58.

4 史忠植. 智能主体及其应用[M]. 北京: 科学出版社, 2000.