长尾理论视角下基于 DCA 的网络自助出版推荐系统①

刘晨晨，徐一新

(复旦大学 文献信息中心，上海 200433)

摘　要：随着自助出版系统文本规模的迅速增长，选用合理的推荐技术有利于“长尾”文本的发掘和价值实现。针对自助出版文本，设计了基于有向图的聚类算法 DCA（Directed Graph Clustering Algorithm），将聚类看成是确定对象的过程，根据词间信息传递量的大小选定特征词集对文本进行聚类。为改善“长尾”文本聚类的有效性，文中所述系统设置了浮动相似度阈值及推荐公共池。实验结果表明，较之 K-Means 算法，该算法有较强的自适应性和通用性，能有效地运用到自助出版文本的个性化推荐系统领域。

关键词：个性化推荐技术；自助出版系统；长尾；DCA 算法

Recommendation System Based on DCA in Web Self-Publication

LIU Chen-Chen, XU Yi-Xin

(Document and Information Center, Fudan University, Shanghai 200433, China)

Abstract: To deal with the self-publication system’s huge text scale and speedy increasing, a right recommendation technology is helpful to realize the market value of “Long Tail” books. To deal with this issue, Directed Graph Clustering Algorithm is presented. Regarding clustering as the process of objects identifying, key words’ election depends on how much information they transfer in context. Moreover, to improve the efficiency of “Long Tail” texts’ clustering, a floating threshold and a sharing pool are set. Finally, experimental results comparing the K-Means algorithm prove that this clustering algorithm based on the directed graph is self-adaptive and effective.

Key words: personalization recommendation technology; self-publication system; the Long Tail; DCA

1　前言

1.1 自助出版和长尾理论视角综述

2006 年，Blurb 公司推出一款免费的编辑软件 Booksmart 并且提供按需印刷的服务，美国即开始兴起了由作者自己编辑、制作、印刷，并承担全部出版费用的出版形式，这一形式让网络图书市场达到了前所未有的规模[1]。早在 2004 年，Chris Anderson 在《长尾》一文中就提出了“长尾理论”，指出电子商务的高速发展使“热门商品”之后排成了一个长长的“尾巴”，而这个“尾巴”所占市场份额的汇聚足以与那些“热门”商品相抗衡[2]。网络自助出版的经济零门槛，使这一市场的“长尾”效应尤为明显。

因此，如何利用相关技术让“长尾”图书实现价值以及如何使新增文本能够快速得到归类整合都是自助出版模式给计算机系统带来的挑战。

1.2 个性化系统推荐技术综述

目前，个性化推荐系统的实现技术主要有关联规则技术[3]，协同过滤技术[4]和内容过滤技术[5]。关联规则技术和协同过滤技术只能通过研究读者行为的相似性挖掘重复路径来形成推荐，并不能有效帮助读者发现相对冷门处于“长尾”位置的书籍。同时，不管是运用哪种推荐技术，一个能够自动调整分类体系的后台文本库无疑可以大大提高系统的推荐性能。因此，本文旨在采用有效的聚类技术，从内容出发实现文本库的自适应结构调整和“长尾”挖掘。

对非结构化文本的聚类多以向量空间模型 VSM（Vector Space Model）[6-8]为基础。VSM 模型以赋予文本中的某些关键词条以权值构成矢量来刻画文本内

① 收稿时间:2010-12-05;收到修改稿时间:2011-01-15

容，方法简单易操作，但是该模型依赖于关键词条的选定并仅根据词形特征匹配来确定权值；而需要预先确认聚类簇的 K-Means 算法(Baxter 等人,2006)，需要预先设定高频词和低频词的 I-Match 算法(Chowdhurry 等人, 2002)，需要遍历所有文档的 Single-Pass 算法(Cios 等人,1998)，将句式结构作为特征的 SpotSigs 算法(Martin Theobald 等人,2008)及设定边界以短语为聚类特征的 STC 算法(Hung Chim 等人,2007)都不能满足自助出版系统高速的增长规模和对自适应的要求。

因此，本文借鉴 InfoSigs 算法[9]的思想，提出 DCA 算法（Directed Graph Clustering Algorithm）。InfoSigs 算法主要是通过构造信息传递的有向无环图以实现 Web 对象的细粒度聚类。由边和节点构成的有向图能够更好的适用于文本的高速增长，因为系统的更新只需要加入节点修改节点指针即可。DCA 采用图论法，将聚类看成是确定对象的过程，权值的高低取决于该词汇辨别对象能力的强弱，关键词的权值取决于其在一系列词群中的信息传递量。同时，结合出版系统特性，根据同一个词出现在文本不同位置所能代表文本的重要性大小赋予不同的位置权重，以改进算法的推荐精度。

2 基于DCA聚类的个性化推荐

2.1 系统构建

从信息论的角度分析，两个不同的特征词在文本数据集中存在一种类似树状的概率层次关系，在一个词出现的记录中另一个词有很大概率同时出现，即一个对另一个在辨认对象方面起到了信息补充的作用；可是噪音词汇对于其他的特征词不存在上述概率层次关系[10]。如“上海旅游 东方明珠塔 优惠”，其中，“东方明珠塔”可以具体化“上海旅游”，而“优惠”则是对分辨对象不起作用的噪音词汇。

根据上述特点，本文所构建基于图论聚类的个性化推荐系统由以下几个部分组成：

① 数据预处理：找出能够辨别不同文本的特征描述，把这种描述转化为计算机能加以处理的记录，包括词干提取，停用词过滤还有文本分割等处理技术。考虑到自主出版系统的文本规模，基本预处理数据为标题、内容简介和目录。值得注意的是，大部分小说和诗歌的标题并不能简单作为特征词加以处理，因为其标题往往无法代表其内容。因此，在这一阶段，系统还将针对小说诗歌等的文本类型特点，做一个文本学习模型，以完成书本内容的预标注形成预处理数据集。

② 相似度计算：本文先构建有向无环图计算各节点权重，根据权重确定特征词后，在各节点设置倒排索引作为 Jaccard Coefficient[11]相似度计算的参数。

③ 记录簇合并：设定浮动的相似度阈值将文本集反复聚类，从而在有效聚类的基础上整合“长尾”文本。

④ 个性化推荐：自动匹配查询特征词与聚类簇聚类中心特征词以得到推荐结果。

2.2 核心算法描述

2.2.1 算法形式化定义：

定义 1. 设文本集 D，词集合 W，记录集合 R，$R = \{r \mid r = (w_1, w_2, ..., w_k), \forall 0 < i < j < k, w_i, w_j \in d_k, w_i, w_j \in w, \text{且} tf(w_i) \le tf(w_j)\}$。$tf(w)$ 表示词 w 在集合 D 中出现的频率，即词频（term frequency, TF）。

定义 2. 有向无环图 G=(V,E)，节点集合 V 即集合 W，边集合 $E = \{(w_i, w_{i+1}) \mid w_i, w_{i+1} \in r\}$。每条记录 r 对应 G 中一条有向路径，由于每个词在数据集中词频确定，且呈正序排列，所以 G 中不存在环。

2.2.2 节点权重计算

权重是用来衡量特征词对文本内容代表性大小的数值，权重越大则该词越能代表文本的内容。在计算之前，首先要根据各文本预处理数据集的词频（见定义 1）构造有向无环图（定义 2），然后再计算各节点的权重。Shannon 从热力学定律出发，使用数学语言阐述了概率与信息不确定性的关系[12]，提出了熵的概念。信息熵即信息中排除了不确定性之后的平均信息量。因此，在文本中，根据特征词群出现的概率可以算出各词的平均信息量，并得到其权重。

$$H_r(w) = -\sum_{v \in W} \frac{P(v|w)}{P(w)} \ln\left(\frac{P(v \mid w)}{P(w)}\right) \times \sum_{i=1}^{3} L_i(w) \qquad (1)$$

其中，$H_r(w)$ 表示特征词 w 的信息熵，c 表示在出现特征词 w 的前提下，出现特征词 v 的条件概率。

同时引入位置系数 $\sum_{i=1}^{4} L_i(w)$ 根据特征词在文中出现的位置进行权值的调整，其中 i 为该词的位置标识。对于非小说诗歌类文本，位置系数主要是三个，即 L_1、L_2 和 L_3，分别代表标题、摘要、目录。位置系数计算公式如下：

$$L_i(w) = \begin{cases} 2^{-i}, & w \in L_i \\ 0, & else \end{cases} \qquad (2)$$

而对于小说诗歌类文本，需要引入在数据预处理

阶段产生的预标注信息，位置系数为：L_1，L_2，L_3 和 L_4，分别代表预标注、标题、摘要、目录。

文本标题与摘要的语言高度概括性往往在一句话的范围内便能表达相对完整的概念。因此，特征词距离的远近对信息量显然存在影响。本文算法引入系数 K(0< k < 1),表示节点距离的间隔增大导致的信息量耗损。因此信息量的最终公式为:

$$I_r(w)=-\left(\begin{array}{l}\sum\limits_{m\in W} P\left(\frac{P(m\mid w)}{P(w)}\right)\ln\left(\frac{P(m\mid w)}{P(w)}\right)\\ +\sum\limits_{n\in W} P\left(\frac{K\times P(n\mid w)}{P(w)}\right)\ln\left(\frac{K\times P(n\mid w)}{P(w)}\right)\end{array}\right)\times L_i \tag{3}$$

$I_r(w)$ 为 w 的信息量，m 表示与 w 处在同一句话中的近距离特征词，n 则表示与 w 不在同一句话中的远距离特征词。

将 w 的信息量结果归一化得到 w 权重的最终公式为：

$$g(w)=\frac{I_r(w)-Min(I)}{Max(I)-Min(I)} \tag{4}$$

其中，$g(w)$ 表示特征词 w 的权重，$Max(I)$ 为信息熵的最大值，$Min(I)$ 为信息熵的最小值。

2.2.3 相似度计算与记录集合并

在确定了特征词权重后需要计算文本相似度形成最终聚类结果。首先，根据权重对各文本特征词排序，考虑到系统效率，选取前十个词构造再此构造一个有向无环图 G′。在图 G′中，遍历数据集，建立各节点的倒排索引, 即出现了该各特征词的文档号集合。形式化定义为：

定义 3. 节点的倒排索引 $F_i=\{f\mid \forall f, f\in D, v_i\in f\}$。

之后，在图 G 的基础上，构建相似度连通图集合 SG，该图节点仍为各特征词。SG 的形式化定义如下：

定义 4. 相似度连通图 SG=(E,V)，节点集合 V 即集合 W，边集合 $E=\{\text{edge}^{'(v_i\to v_j)}\mid w_i, w_j\in w\}$。引入 Jaccard 系数，边集合 E 的计算公式为：

$$\text{edge}^{'(v_i\to v_j)}\leftarrow\begin{cases}1, if\ \dfrac{|F(v_i)\cap F(v_{i+1})|}{|F(v_i)|+|F(v_{i+1})|-|F(v_i)\cap F(v_{i+1})|}\ge\omega\\ \varnothing, else\end{cases} \tag{5}$$

其中，F(n)是节点 n 的倒排索引文本号集合，是一个相似度阈值。计算公式为 Jaccard 系数的二值权重计算方案。至此，图 G′被分解为连通子图集 SG_n，每个子图表示一个聚类簇，簇中节点的倒排索引并集即为这个聚类簇的文本记录集合。可是，这些聚类簇包括了许多冗余的文本，因此需要进一步去重。

首先，根据文本在各特征词倒排索引中出现的次数即频率进行排序。定义如下：

定义 5. 设文本集合 D，聚类产生记录簇集 C，C= $\{c|c\in d\}$,其中，C 为 F(v)的集合，任意一个连通子 SG_i 图对应一个 c_i。$tf(d_j^i)$ 表示文本 d_j 在集合 c_i 中出现的频率。对于 $d_j^i\in c_i$：

$$tf(d_j^i)=Count_{d_j^i}(F(v_1)\cup F(v_2)...\cup F(v_n)) \tag{6}$$

然后把同时出现在不同簇的文本 d_j 根据 $tf(d_j^i)$ 的大小进行去重最后得到聚类簇集合 CL。

值得提出的是，聚类的最后结果必然产生一些不属于任何簇的离散文本，即如前所述的“长尾”，本文的解决方案是系统自动降低相似度阈值，将这些文本重新聚类。而对于确实无法聚集的文本，系统将设定一个公共池，其中的文本以若干特征词代替，根据特征词的主题类别进行推荐。

2.2.4 个性化推荐

定义 6. 聚类簇中心特征词集合 $w_k^{cl_i}$，其中，k=1,2…n, $w_k^{cl_i}$ 为聚类簇 cl_i 所有文本中出现次数最多的特征词。同样考虑到系统效率，此处取 TOP10。

当读者在系统中输入查询条件时，系统将同样对查询条件进行预处理，取得查询特征词后与聚类簇聚类中心特征词相匹配。确定推荐簇后，根据中心特征词在各文本中权重的大小进行排序并依次推荐。由于本系统采用根据相似度阈值分解有向无环图的方法，而图一旦构建更新相对稳定和简单，因此本系统可以实现阈值的个性化设置，根据读者的偏好选择聚类的大小从而形成推荐集。但是，为了更有针对性地对读者形成推荐，事先系统仍然需要对阈值的设置进行最小控制，以避免推荐结果集过大。

2.2.5 DCA 算法描述

算法 1 // 据特征词集合构造有向无环图

输入：文本集合 D,和词集合 W

输出：有向无环图 G

1) Set G $\leftarrow\varnothing$

2) Set R $\leftarrow\varnothing$

3) For all the words $w_1, w_2, ..., w_k\in d_j$ do

4) Get the set R based on the sequence of tf(w) // 根据各词在文本中出现的频率排序得到记录集 R

5) For each item $w_j, r_j\in R$

6) Set $V \leftarrow w_j$ // 节点集合 V 即集合 W

7) Set $E \leftarrow \text{edge}(\text{v}_i \rightarrow \text{v}_{i+1})$ // 每条记录 r 对应 G 的一条路径

8) End for

9) End for

10) For each item $v_j \in V$

11) Get $I_r(v_j)$ using formula (3) // 计算节点信息量

12) Get g(v_j) using formula (4)// 计算特征词权重

13) Get F_i // 构建节点 v_i 的倒排索引

14) End for

15) Return G

算法 2 // 基于有向无环图分解的聚类算法

输入：有向无环图 G,各节点的倒排索引集 F, 相似度阈值 ω

输出：聚类簇集合 CL

1) For each node pair $v_i, v_j \in V$

2) Get $\text{edge}^{'(v_i \rightarrow v_j)}$ using formula (5) // 以节点倒排索引为计算参数分离图 G 得到连通子图集

3) Get CL using formula(6) // 聚类文本去重

4) If $d_j \notin CL$

5) ω＝ω *0.9 // 重置相似度阈值

6) Redo (3) // 重新计算相似度

7) Else Put d_j in Sharing Pool // 推入公共推荐池

8) End for

9) Return CL

3 实验

为验证本文 DCA 算法的有效性，与基于向量空间模型的经典 K-means 算法[13]进行了分析比较。K-means 算法的主要思想是先任意选定 K 个对象为初始聚类中心，然后把其他对象根据与这个 K 个对象的相似程度进行聚类。本文实验数据来自于笔者设计的“网上书城”数据库，该库图书信息均来自亚马逊、卓越及当当等大型网上书城。笔者根据各类别选择了 200 个文本，抽取其题目、摘要和目录，并事先对数据集进行了人工聚类，作为标准聚类结果。由于本文算法与 K-means 算法得出的聚类个数不相同，为了方便比较，本实验采用聚类准确度系数和聚全率系数分别对两个算法进行综合评价。同时本实验还将数据集分组，分批测试，以观察两个算法随数据集增大的效率变化。

在聚类簇确定后，为了验证本文针对“长尾”文本所设计推荐公共池的效用，笔者还设计了一个简单地测试用户满意度的实验。测试对象为在校生 10 人，要求在三天内浏览系统不少于 6 个小时。实验在提供公共池推荐和不提供这两种情况下分析用户的满意度指标。

算法的开发工具是 VC++6.0，所有试验在内存为 2G，CPU 为 Intel Core™21.8GHz，操作系统为 Windows 2000 Server，数据库管理系统为 SQL Server 2000 的 PC 上进行。

3.1 实验评测指标构造

聚类准确度（Pre）：以标准聚类结果为依据，每个标准类的文本集合分布于算法的各个聚类簇中，聚类准确度就是计算这些散布在各个聚类簇中的数据集占其所在簇的比率，为了便于分析，只计算所占比重最大的类。

$$\text{Pre} = \frac{\sum_{i=0}^{n} |\text{CL}_i \cap \text{S}_j^i|}{\sum_{i=0}^{n} |\text{CL}_i|} \times 100\% \quad (7)$$

聚全率(All)：以标准聚类结果为依据，计算占聚类簇比重最大的相似文本集与标准聚类结果该类文本数量的比率。

$$\text{All} = \frac{\sum_{i=0}^{n} |\text{CL}_i \cap \text{S}_j^i|}{\sum_{i=0}^{n} |\text{S}_j^i|} \times 100\% \quad (8)$$

其中，CL_i 表示聚类簇 i 的文本数量，S_j^i 表示在聚类簇 i 中比重最大的相似文本集所对应的标准聚类 j 的数量，$|\text{CL}_i \cap \text{S}_j^i|$ 表示聚类簇 i 中属于类 j 的文本数量，当 $|\text{CL}_i \cap \text{S}_j^i| = 1$ 时，该簇系数设 0。

客户一次推荐满意度：对用户访问的页面中推荐页面所占比例来刻画用户对这一次推荐的满意度。

$$Stf_{step} = \frac{|P_{s_i} \cap Q_{s_{i+1}}|}{|Q_{s_{i+1}}|} \quad (9)$$

其中，$|P_{s_i} \cap Q_{s_{i+1}}|$ 表示在推荐集合中出现的，并在下一个执行步被访问的页面的个数，$|Q_{s_{i+1}}|$ 表示下一个执行中被访问的页面总数。

3.2 实验结果分析

实验将初始数据集随机分为两组，两组均为 100 条文本标题与摘要记录。首先，将第一组记录集合的聚类结果进行分析，将阈值 ω 设为 6.5，K 设为 8，结果如图 1。

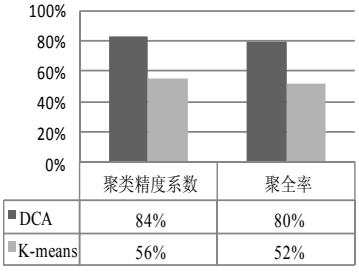

图 1　两种算法的平均性能比较

实验的第二步是将剩下的 100 条记录分成 3 组，记录数分别是 10,30,60，依次添加到系统中，测试两个算法的自适应性和健壮性。结果如图 2 和图 3。

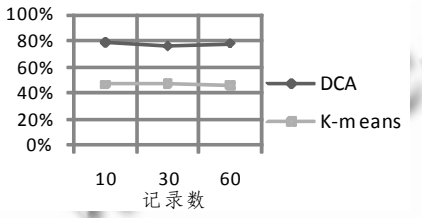

图 2　分组实验结果-聚类精度比较

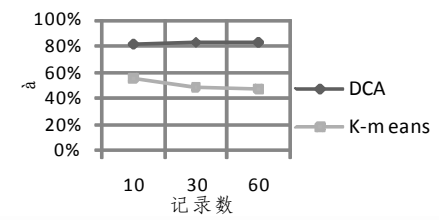

图 3　分组实验结果-聚全度比较

实验的第三步是累加各次推荐满意度后根据推荐次数平均算出总体满意度。结果如图 4。

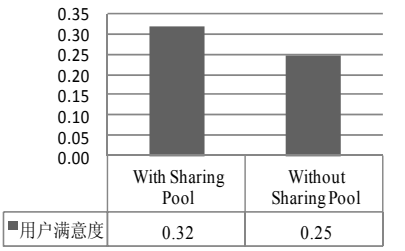

图 4　推荐池用户满意度验证

由图 1 可知，DCA 算法的聚类精度相比传统的 K-means 提高了 21%，聚全度提高了 19%。结果显示，基于有向图聚类的 DCA 算法能够有效的识别文本描述对象，对于文本聚类有较高的准确性。

从图 2 与图 3 可以看出，在第一次聚类的基础上进行动态再聚类使算法准确性有所降低，这是因为随着记录的不断增加，分类变得更加杂乱，加大了聚类的难度，且一些小的集合未能纳入测评系数的计算范围。但是在这个过程中，DCA 的准确性的降低幅度不如 K-means 大，尤其是追加记录集由 30 条记录增加到 60 条时，前者基本保持了平衡，而后者出现了下降趋势明显。且在每部分实验中，DCA 算法的准确性均优于 K-means 算法。图 4 的结果验证了推荐公共池的效用，虽然增长幅度不太显著，但值得注意的满意度的增长都来自于公共池的推荐。经过分析可知，利用有向图构造数据集描述增强了系统的自适应性和通用性，可以有效地运用到个性化推荐系统领域。

4　总结

长尾理论揭示了电子商务技术的发展对市场利润分配的影响。电子设备的高速发展促进了自助出版的成功运作，也给计算机系统带来了新的挑战。据此，本文提出了聚类算法 DCA ，实验结果证明，基于有向图数据结构的聚类不但能提高聚类效率，而且在面对文本大量增加的情况下仍然能够保证良好的性能。同时，推荐公共池的设置也为“长尾”文本展示提供了一个有效方案。

在未来的研究工作中，我们将关注于算法的不断改进，如优化推荐公共池规则、加强算法抗噪音干扰的能力、改善算法的时间复杂度以及对算法各参数阈值的分析确认等。同时，我们还会以该算法为基础，建立一个完整的书库推荐系统并实现与网络用户的交互，以此来搭建一个真正意义上的网络自主出版系统推荐模型。

参考文献

1 刘肖.网络自助出版模式研究.出版发行研究, 2007,(11):42–45.

2 Anderson C. The Long Tail: Why the Future of Business is Selling Less of More. New York: Hyperion, 2006.

3 Adomavicius G, Tuzhilin A. User profiling in personalization applications through rule discovery and validation. Lee D, Schkolnick M, Provost F, et al, eds. Proc. of the 5th International Conference on Data Mining and Knowledge Discovery. New York: ACM Press, 1999. 377–381.

(下转第 105 页)

根据上述运算所得特征权重及用户需求输入，按基于实例的可配置产品功构映射过程中需求特征与实例特征相似性计算方法，可得用户功能需求特征与实例库样本特征相似性如表 6 所示，并根据∑ω*d 可得与用户需求特征最相似的样本实例 3。设计人员根据样本实例 6 的结构特征，按直接转换、置换法、改造法或基于框架的方案变换对样本 6 进行修正。设计人员完成样本修正后，将修正的实例加入实例库中，从而实现系统的自我学习任务。实例的学习过程包括实例的评价和修正，即评价和确认是否进行实例更新。具有较高评价的实例为成功实例，具有较低评价的实例为失败实例。实例的更新依赖于当前实例与库中实例的相似性，其主要包括更新信息内容确定、更新形式确定、实例索引及实例存储。

5 结论

(1) 分析了目前基于实例推理方法中权重制定的主观性对推理结论的影响。

(2) 提出了基于神经网络的 CBR 权值学习实现方法及输入特征转换机制，在此基础上给出了基于神经网络算法和 CBR 的需求映射实现过程。

(3) 将上述方法应用于移动工作台功能特征映射取实现过程中，实践证明上述方法的有效性。

参考文献

1 Kwang HI, Sang CP. Case-based reasoning and neural network based expert system for personalization. Expert Systems with Applications,2007, 32(1): 77−85.

2 Segee BE, Carter MJ. Fault tolerance of pruned multilayer networks. Proceedings of international joint conference on neural networks, 1991, II: 447−452.

3 王世伟.基于知识的产品配置建模、演化及其应用研究[博士学位论文].杭州:浙江大学,2004.

4 谢清.面向方案设计的可配置产品功能--结构映射原理、方法及关键技术研究[博士学位论文].杭州:浙江大学,2007.

(上接第 30 页)

4 Badrul S, Karypis G, Konstan J, Riedl J. Analysis of recommendation algorithms for e-commerce. Proc. of EC'00. Minneapolis, 2000.

5 Mobarshel B, Cooley R, Srivastava J. Automatic personaliztion based on Web usage mining. Communications of ACM, 2000,43(8):142−151.

6 Baeza-Yates RA, Ribeiro-Neto B. Modern Information Retrieval. MA: Addison-Wesley, 1999.

7 Aas K, Eikvil L. Text Categorisation: A Survey. Norwegian: ACM Computing Center, 1999.

8 Xu R, Wunsch D. Survey of clustering algorithms. IEEE Trans. on Neural Networks, 2005,16(3):645−678.

9 Sheng ZH, Wu Y, Jiang JH, Shou LD, Chen G. InfoSigs: A Fine-Grained Clustering Algorithm for Web Objects. Journal of Computer Research and Development, 2010, 47(5): 796−803.

10 Zhao Y, Wang XL, Liu BQ,. Fusion of clustering trigger-pair features for POS tagging based on maximum entropy model. Journal of Computer Research and Development, 2006,43(2):268−274.

11 Theobald M, Siddharth J, Paepcke A. SpotSigs: Robust and efficient near duplicate detection in large Web collections. Proc. of the 31st SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 2008.563−573.

12 Shannon CE. A Mathematical Theory of Communication. The Bell System Technical Journal, 1948,27:379−423, 623−656.

13 Gu L, Baxter R. Decision models for record linkage. LNCS 3755: Data Mining AusDM. Berlin: Kluwer Academic Publishers, 1998.