# 基于图的查询日志实体别名抽取方法*

石贝，孙乐，韩先培

中国科学院软件研究所 北京 100190

E-mail: {shibei, sunle, xianpei}@nfs.iscas.ac.cn

**摘要：**实体的别名是指同一个实体的不同名称。传统的别名抽取方法存在训练语料构建困难和时效性差这两个问题。针对这两个问题，本文提出了一种基于图的查询日志实体别名抽取方法。该方法利用查询日志的上下文信息和查询链接信息，构建了二层图（包括别名候选图层和查询链接图层），并通过随机游走算法对图中的候选别名进行排序。实验结果表明：1)该方法达到了 71.8%的准确率，证明该方法可行有效。2)使用查询链接信息进行别名抽取优于使用上下文信息进行别名抽取。这两种信息的结合能获得更好的别名抽取效果。

**关键词：**查询日志；别名抽取；

**中图分类号：**TP391　　**文献标识码：**A

# Graph Based Alias Extraction using Query Log

Bei Shi, Le Sun, Xianpei Han

Institute of Software, Chinese Academy of Sciences, Beijing 100190

E-mail: {shibei, sunle, xianpei}@nfs.iscas.ac.cn

**Abstract:** The alias of entity means the different names which refer to the same entity. Traditional alias extraction methods often have two problems: 1) the difficulty of constructing training corpus; 2) the lack of timeliness. To resolve the two problems, this paper proposes a graph based alias extraction method using query log. This method uses context information and query-link information, constructs a two-layer graph (including the candidate alias layer and the query-link layer) and sorts the alias using random walk algorithm. The experimental results show that: 1) our method achieves the accuracy of 71.8%, which proves our method is effective. 2) Using query-link information outperforms the method which uses context information and the combination of this two type's information improves the performance.

**Key Words:** query log; alias extraction;

## 1　引言

自然语言的多样性决定同一实体能通过不同的别名来表达。实体的别名是指同一个实体的不同名称。它主要包括缩略语，曾用名，拼音和其他语言的翻译等。比如“人人网”的别名有“人人”，“校内网”，和“renren”等。别名抽取是指输入一个实体的名称，抽取并返回代表该实体的其他名称。别名抽取的相关研究是自然语言处理的重要课题。它可以用于知识库的构建，机器翻译，问答系统，信息检索和实体链接等具体应用中。比如，信息检索面临的一个重要问题就是相同的实体采用不同名称从而造成检索召回率的降低。使用抽取后的别名进行查询重构可以解决这个问题。

* **基金项目：**基于查询语义分析与推理的隐式相关反馈检索模型研究（61272324）；

**作者简介：**石贝，男，硕士，主要研究方向：信息检索；孙乐，男，研究员，主要研究方向：自然语言处理；韩先培，男，副研究员，主要研究方向：自然语言处理。

目前别名抽取主要是利用已有的语料库（采用别名、原名的平行语料库或者经过人为分词和词性标注的语料库）提取候选别名、原名集合，再利用别名、原名的上下文模板等字对齐规则进行搜索匹配[1]，或者采用机器学习的方法进行选择匹配对，最后输出正确的原名、别名对。

这类方法主要存在两个问题：

1）语料库构建困难

由于别名的多样性（包含拼音，缩略语，翻译，曾用名），构建高覆盖率的别名平行语料十分困难。比如，“中国建设银行”的别名包含“建设银行”，“建行”，“CBC”和“Jian She Yin Hang”等。我们没有找到包含“中国建设银行”上述所有别名的平行语料。

2）时效性差

社交网络时代的来临让人与人在网上的沟通更加频繁，从而使自然语言的经济性原则得到了充分利用。“Weibo”，“KFC”，“神九”等各种新别名出现越来越频繁。Web 信息的日益膨胀使平行语料的构建速度远远赶不上别名的产生速度。如何不通过平行语料，及时地抽取别名成为了一个挑战。

随着互联网信息的不断产生，利用用户生成的内容进行别名抽取成为解决上述两个问题的关键。本文提出了基于图的查询日志别名抽取方法。首先，本文基于查询和链接对应的点击信息，进行一次迭代，获取候选查询集合。然后，本文基于<别名—模板>对和<查询—链接>对构成二层图，采用随机游走方法对候选别名进行排序，抽取出权重较高的名称作为实体的别名。

本文安排如下：

第二节介绍了别名抽取的相关方法；第三节首先介绍了算法的框架，然后详细介绍查询日志中两大类影响别名抽取的信息（上下文信息和查询链接信息），并就各类信息的实际意义及计算方法进行详细说明，最后综合利用这两个信息特征，提出一个基于二层图的别名抽取框架；第四节用实验对比各类信息的性能差异，并证明本文做法有效可行。第五节对本文进行总结，分析并提出下一步的工作重点和研究问题。

## 2 相关工作

由于别名包括缩略语等形式，所以缩略语抽取的相关工作与别名抽取密切相关。Zhu et al. 针对单字人名、地名简称，构建了基于分类器的预测模型 [2] 。李斌等对汉语单字国名采取了统计评分法进行识别 [3]。Chang 和 Lai 使用人工标注的源短语、缩略语的平行语料库作为训练数据，然后利用 HMM 来提取缩略语、源短语对 [4]。Chang 和 Teng 提出了基于 HMM 的概率恢复模型(SCR)，用于将缩略语扩展为源短语[5]。崔世起利用生语料使用重复串搜索技术和词性过滤，必要时加入人工干预，自动提取缩略语和源短语对[6]。武子英利用上下文语义信息，基于余弦相似度自动抽取汉语缩略语[7]。Li 根据缩略语与源短语的共现现象，使用字对齐规则进行自动提取缩略语（仅处理单一类型的缩略语）[8]。上述方法尽管目前只用于缩略语抽取，但是也可以用于别名抽取的工作中。但是这些方法具有选用的语料库时效性较差，规模较小，需要人工干预，且只解决缩略语抽取问题等缺陷。

Bollegala 等人采用搜索引擎获得候选人名别名的集合，然后利用 SVM 分类器进行人名别名抽取[9]。Bhat 等人采用 LSA 方法，利用不同的别名周围具有相同的上下文特征，进行别名抽取，但该方法有时效性低，运算量大等缺点[10]。同上述方法相比，本文方法不仅使用了上下文信息，还使用了查询链接信息，提高了抽取的准确率。

对于上述方法需要平行语料，时效性低等缺点，本文提出了使用用户查询日志，自动抽取候选查询，然后利用上下文信息和查询链接点击信息构建二层图，再使用随机游走算法对图中的候选别名进行排序的方法。该方法不需要任何标注数据和人工干预，并具有很好的时效性。

## 3 基于查询日志的别名抽取算法

### 3.1 算法框架

在输入原名 $e$ 后，别名抽取的目标是向用户返回查询日志中实体的别名$\{a_1,a_2,...,a_p\}$。为了便于展示，全文将通过抽取“人人网”的别名这个例子对我们的方法进行描述。相关定义如下：

- 输入原名 $e$：“人人网”；
- 已知由查询记录$\{r_1,r_2,r_3,...,r_s\}$构成的文档集合 $R$。每一条记录包含查询（用 $q$ 表示）和查询对应的点击链接（用 $l$ 表示）。比如，其中一条查询记录为“北京大学 http://www.pku.edu.cn”；
- 目标集合$\{a_1,a_2,...,a_p\}$：“人人网”的别名构成的集合。比如，“人人”，“校内网”，“xiaonei”等别名所构成的集合。

在查询日志中，我们观察到别名的特征主要包括两类：

（1） 和原名具有相同的上下文。比如对于原名“人人网”，查询日志包含大量查询“人人网首页”，同时查询日志也包含大量查询“校内网首页”。因此，“人人网”和“校内网”包含相同的上下文“*首页”。

（2） 别名所构成的查询和原名所构成的查询被用户点击到同一链接。对于查询“人人网地址”，其指向的链接为“http://www.renren.com”。对于查询“校内网地址”，其指向的链接也为“http://www.renren.com”。

因此，利用这两类特征，本文提出的基于图的查询日志别名抽取算法的框架如图 1 所示。

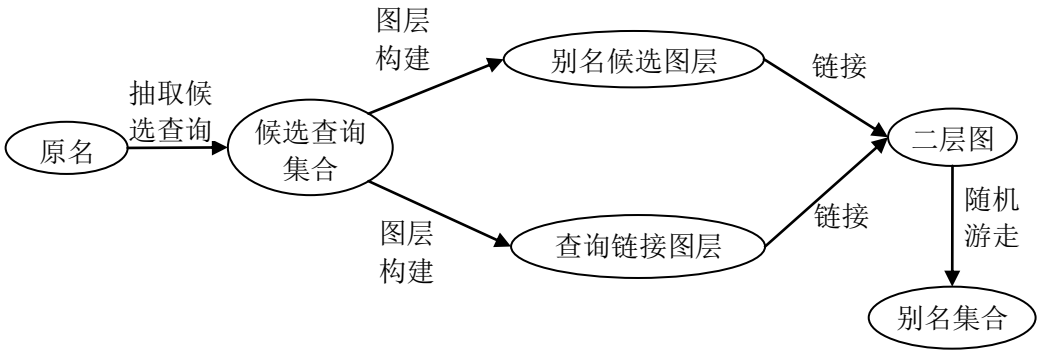


图 1 基于图的查询日志别名抽取算法框架

### 3.2 抽取候选查询集合

实验中查询日志包含冗余查询，数据量大，因此在别名抽取前需要对查询日志进行过滤，得到查询记录的子集——候选查询集合 $Q_c$。候选查询集合是有可能包含原名和别名的查询所构成的集合。我们的假设是，包含别名的查询和包含原名的查询至少指向一条共同的点击链接。基于此假设，抽取“人人网”的候选查询集合 $Q_c$ 的步骤如下：

1) 将“人人网”和查询日志中的查询逐条匹配。如果“人人网”是查询 $q$ 的子串，则将 $q$ 加入集合 $Q_0$ 中。$Q_0$ 为包含原名“人人网”的查询所构成的集合。
2) 对 $Q_0$ 中的每一个查询 $q$，获得它对应的点击链接 $c$（每一个查询对应的点击链接可能有多个），将 $c$ 加入链接集合 $C_0$ 中。
3) 对 $C_0$ 中的每一条链接 $c$，获得 $c$ 对应的查询 $q'$，将 $q'$加入候选查询集合 $Q_c$ 中。

如图 2 所示，我们利用了查询-链接信息，生成了候选查询集合 $Q_c$。$Q_c$ 中的查询有可能包含别名。

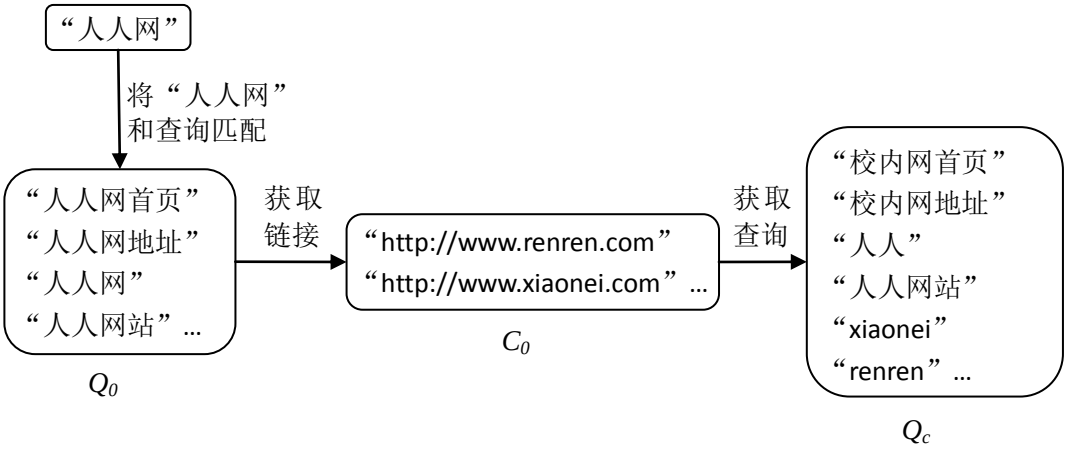


图 2“人人网”候选查询集合的生成过程

### 3.3 二层图构建

在获取候选查询集合 $Q_c$ 之后，我们需要抽取出 $Q_c$ 中的查询所包含的别名，并对其排序。在此步骤中，本文首先构建别名候选图层，然后构建查询链接图层，然后将这两个图层进行链接，得到二层图。最后采用随机游走算法进行排序，得到最终结果。

#### 3.3.1 别名候选图层的构建

我们观察到对于原名和别名，它们很可能共享相同的上下文。我们的假设是，如果一个查询和包含原名的查询有相同的上下文，那么这个查询可能包含别名。在得到候选查询集合 $Q_c$ 后，本文使用基于模板的 Bootstrapping 算法生成<别名-模板>对和<模板-别名>对，然后构建别名候选图层。构建过程如下：

1) 将原名 $e$ 加入命名集合 $N$ 中。将模板池 $W$ 置空。
2) 分析 $Q_c$ 中的每个查询 $q$，若 $q$ 包含命名集合 $N$ 中的元素 $n$，则抽取 $n$ 的上下文，

生成模板 *w*。其中，在 *q* 的句首和句尾添加 “<s>”和“</s>”标签作为开始标记和结束标记。比如对于原名“人人网”，如果 $Q_c$ 中存在查询“人人网首页”，则生成模板“<s>*首页</s>”。将模板 *w* 加入模板池 *W* 中，同时记录<别名-模板>关系。为提高模板的有效性，减少随机事件的影响，本文在这一阶段过滤掉模板池中只出现过一次的模板。

3) 依次取出模板池 *W* 中的模板 *w*。对于 $Q_c$ 中的每一条查询 *q*，若 *q* 匹配模板 *w*，则抽取出候选别名 *a*，并将 *a* 加入 *N* 中，同时记录<模板-别名>关系。

4) 重复 2 步和 3 步，直至没有新的元素加入 *N* 中。

利用上述步骤得到的<别名-模板>对和<模板-别名>对，对集合 *N* 和 *W* 构建二分图。对每一个<别名-模板>对和<模板-别名>对，在图中添加相应的边来连接对应的别名节点和模板节点。如图 3 所示。

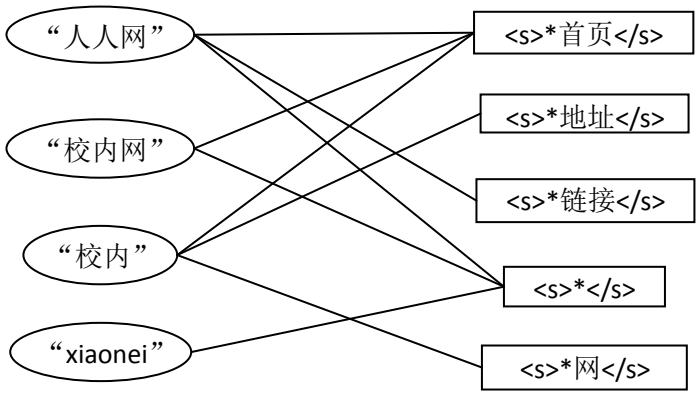


图 3 别名候选图层的构建示例

### 3.3.2 查询链接图层的构建

为了验证使用查询日志进行别名抽取的有效性，我们对查询日志进行分析。我们抽取出查询日志中，某些 *URL* 链接对应的查询，并按点击次数进行排序，如表 1 所示。

| URL | Query | Count |
|---|---|---|
| http://www.renren.com/ | 人人网 | 5714 |
| | xiaonei | 3082 |
| | 校内网 | 1949 |
| | 人人 | 1733 |
| | renren | 1265 |
| http://www.sina.com.cn/ | sina | 4759 |
| | 新浪网 | 3379 |
| | 新浪网首页 | 284 |
| | 新浪 | 131 |
| | 新浪微博 | 99 |

表 1 查询日志中某些 *URL* 对应的查询

从表 1 中我们观察到，在查询日志中，对于同一个链接“http://www.renren.com”或者“http://www.sina.com.cn/”，其对应的查询（“人人网”和“xiaonei”，“sina”和“新浪网”）互为别名。因此，我们得到假设：对于同一个链接，如果它和几个查询有很强的关联，那么这几个查询很可能包含同一实体对应的别名。利用此假设，我们构建查询链接图层的步骤如下：

1) 对 $Q_c$ 中的每一个查询 $q$，将 $q$ 加入查询链接图层中，同时将 $q$ 对应的链接 $l$ 作为节点加入图层中，并添加边<$q$, $l$>
2) 对新添加的 $l$，如果存在 $l$ 对应的查询 $q$ 不在图层中，则将查询 $q$ 添加到图层中，同时添加边<$l$, $q$>

重复上述两步，直至没有新的边和节点加入此二分图中，则构建出如图 4 所示的查询链接图层。

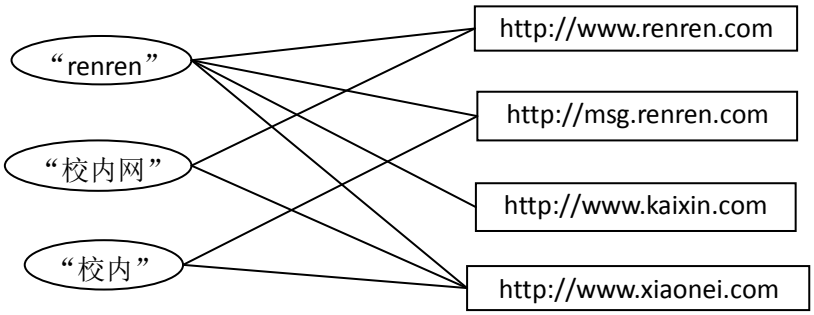


图 4 查询链接图层的构建示例

### 3.3.3 二层图构建

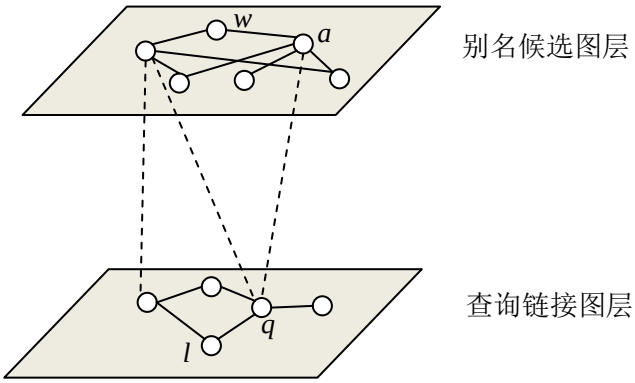


图 5 二层图构建示例

在构建别名候选图层和查询链接图层后，我们构建一个二层图。如图 5 所示，上层为构建好的别名候选图层，$EG=\{EV, EE\}$。$EV$ 是节点集合，包括原名 $e$，候选别名 $a$ 和包围别名的模板 $w$。$EE$ 是边$(ev_i, ev_j)$的集合，$(ev_i, ev_j)$表示节点 $ev_i$ 和 $ev_j$ 之间的边。下层为构建好的查询链接图 $QG=\{QV，QE\}$，$QV$ 是节点集合，节点为查询候选集合 $Q_c$ 中的查询和链

接，*QE* 是下层节点之间的边的集合，每个边用日志中查询和链接的共现来表示。如果用户在查询 *q* 时点击链接 *l*，则在 *q* 与 *l* 之间添加一条边链接。上下两层图通过查询和别名的包含关系连接。对于上层图中的别名 *a*，如果在下层图中存在查询 *q* 包含别名 *a*，则在 *q* 和 *a* 之间添加一条边（图 5 中虚线所示）。

### 3.3.4 随机游走算法

二层图构建完成之后，本文使用随机游走算法计算图中节点的权重[11]，然后对权重排序，返回排名靠前的别名节点。

假设该图中节点的初始权重为：

$$W^0 = \{w_1^0, w_2^0, \ldots, w_n^0\}$$

其中$w_i^0$是编号为$i$的别名，模板，查询或者链接的初始权重。该二层图总共包含 *n* 个节点。则二层图中边的权重可以表示如下：

$$E = \begin{bmatrix} e_{1,1} & e_{1,2} & \cdots & e_{1,n} \\ e_{2,1} & e_{2,2} & \cdots & e_{2,n} \\ \cdots & \cdots & \cdots & \cdots \\ e_{n,1} & e_{n,2} & \cdots & e_{n,n} \end{bmatrix}_{n \times n}$$

其中$e_{i,j}$表示节点 *i* 和 *j* 之间的边的权重。针对不同的节点 *i* 和 *j*，其权重的计算方式分为如下情况：

1) 如果 *i* 是别名或者原名，*j* 是模板，那么我们使用别名 *i* 和模板 *j* 在 $Q_c$ 中的共现次数作为权重。
2) 如果 *i* 是模板，*j* 是别名或者原名，那么我们使用模板 *i* 和别名 *j* 在 $Q_c$ 中的共现次数作为权重。
3) 如果 *i* 是查询，*j* 是点击链接，那么我们使用在查询日志中 *i* 和 *j* 的对应点击次数作为权重。
4) 如果 *i* 是点击链接，*j* 是查询，那么我们使用在查询日志中 *i* 和 *j* 的对应点击次数作为权重。
5) 如果 *i* 是别名或原名，*j* 是查询，那么我们使用 $Q_c$ 中别名或原名 *i* 被查询 *j* 的包含次数作为权重。
6) 如果 *i* 是查询，*j* 是别名或原名，那么我们使用 $Q_c$ 中查询 *i* 包含别名或原名 *j* 的次数作为权重。

然后我们对 *W* 进行迭代更新，如下所示。

$$W^{t+1} = \lambda W^0 + (1-\lambda) W^t \cdot norm(E)$$

其中 *norm(E)*是 *E* 的正规化形式，$W^t$是$W^0$经过 *t* 次迭代之后的权重向量，$\lambda \in (0,1)$是一个自由参数，表示初始向量在更新节点权重时的权重。当迭代次数到达某个限定次数，或迭代结果趋于收敛，则停止迭代更新，作为各节点的最终权重。在本文实验中，默认迭代 50 次。

在进行初始权重赋值时，原名节点赋值为 1，其余节点赋值为 0。在迭代一定次数，得到各节点权重后，对别名节点进行排序，输出排序后的别名列表 *L*。计算列表 *L* 节点权重之间梯度，将最大下降梯度之前的节点进行返回。如图 6 中例子所示，返回虚线之前的节

点。

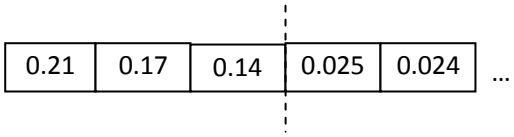


图 6 列表 $L$ 节点返回示例

## 4 实验结果及相关分析

在这个章节，我们评估了本文提出的方法，并将它同三个基准实验进行比较。同时，我们详细分析了实验结果。

### 4.1 数据集

本文实验数据集来自百度搜索查询日志数据（2008 年 10 月），共包含 6515602 个查询。同时，本文共收集了 500 个原名。这 500 个原名主要包含机构名（“中国建设银行”等），品牌名（“索尼”等），和网站名（“新浪”，“人人网”）等。在该实验中，本文基于查询日志对这些原名进行别名抽取，并对抽取结果进行人工评判。

### 4.2 评价方式

在该实验中，我们使用准确率（Accuracy）对实验结果进行评判[12]。对结果集合 $S$，其准确率（Accuracy）为：

$$Accuracy(S) = \frac{|True\ Alias\ in\ S|}{|S|}$$

### 4.3 基准实验

本文的三个基准实验包括：

1) 基于词的上下文相似度的别名抽取方法(ConSim)。该方法抽取原名的上下文，构成向量，然后计算候选词的上下文向量与原名的上下文向量的余弦相似度，然后根据余弦相似度进行排序，进而抽取别名。
2) 仅基于别名候选图层，使用随机游走算法对别名节点进行排序的别名抽取方法(ExtGraph)。
3) 仅基于查询链接图层，使用随机游走算法对查询节点进行排序，直接将排序靠前的查询结果作为别名(QGraph)。

本文提出的基于图的查询日志别名抽取算法被记为(TwoGraph)。

### 4.4 结果与分析

在百度查询日志数据上，四组实验结果如表所示：

| | ConSim | ExtGraph | QGraph | TwoGraph |
|---|---|---|---|---|
| **Accuracy** | 32.2% | 57.0% | 62.8% | **71.8%** |

表 2 四种别名抽取算法的实验结果比较

从上表可以得到如下结论：

1) 四个方法中，我们的方法抽取别名效果最好。别名抽取效果比较结果为：TwoGraph>QGraph>ExtGraph>ConSim。
2) 和 ExtGraph 相比，TwoGraph 在准确率上提高 14.8%。这说明使用上下文信息和查询链接信息进行别名抽取比仅仅使用上下文信息进行别名抽取的效果要好。
3) 和 QGraph 相比，TwoGraph 在准确率上提高 9.0%。这说明使用上下文信息和查询链接信息进行别名抽取比只使用查询链接信息进行别名抽取的效果要好。
4) 和 ConSim 相比，ExtGraph 在准确率上提高 24.8%。这证明了我们使用随机游走算法对构建后的别名候选图层进行节点权重排序的有效性。
5) 和 ExtGraph 相比，QGraph 在准确率上提高 5.8%。这表示查询日志中的查询链接信息比别名的上下文信息更加准确。这可能因为查询日志中查询数目比较多，模板比较繁杂，所得到的上下文信息不如点击信息更加准确。

### 4.5 细节分析

当输入“人人网”后，TwoGraph 系统抽取的别名列表 top10 如表 3 所示。

从表 3 中可以看出，我们实验返回的结果包括了曾用名，拼音，缩略词，URL 和拼写错误。这表明了本文方法保证了别名抽取结果的多样性。该结果也表明了该方法在抽取别名时，也抽取出了一些查询日志中的拼写错误（“xiaone”）。这些拼写错误对系统性能产生了负面影响。

| | | | |
|---|---|---|---|
| **1** | xiaonei | **6** | 校内网登陆 |
| **2** | 人人 | **7** | renrenwang |
| **3** | 校内网 | **8** | 人人网登录 |
| **4** | renren | **9** | xiaone |
| **5** | 校内 | **10** | xiaonei.com |

表 3“人人网”别名列表 top10

## 5 结论及下一步工作

本文针对目前别名抽取需要训练语料，时效性差这两个问题，提出了基于图的查询日志别名抽取方法。本文总结了查询日志的两大类信息（上下文模板信息和查询链接信息），并提出了基于这两类信息的二层图构建算法，然后使用随机游走算法计算候选别名权重，抽取出别名。实验表明：1)我们的方法可行有效，达到了 71.8%的准确率; 2) 使用查询链接信息进行别名抽取优于使用上下文信息进行别名抽取。这两种信息的结合能获得更好的别名抽取效果。下一步工作中，我们将过滤查询日志中的拼写错误，从而降低其对别名抽取结果的负面影响；此外，如何解决查询日志中某些别名的稀疏问题也是下一步的研究方向。

## 参考文献


[1] 刘友强，李斌，奚宁等.基于双语平行语料的中文缩略语提取方法[J].中文信息学报 2012. 26(2): 69-74

[2] Xiaodan Zhu, Mu Li, Jianfeng Gao, et al. Single Character Chinese Named Entity Recognition[C] Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, ACL, 2003.

[3] 李斌，方芳. 中文单字国名简称的自动识别[J]. 计算机工程与应用 2006,42(28):167-176

[4] Jing-Shin Chang and Yu-Tso Lai. A preliminary study on probabilistic models for Chinese abbreviations. [C] Proceedings of the 3rd SIGHAN Workshop on Chinese Language Processing, 2004, 9-16.

[5] Jing-Shin Chang and Wei-Lun Teng. Mining Atomic Chinese Abbreviation Pairs: A Probabilistic Model for Single Character Word Recovery. [C] In Proceedings of the 5rd SIGHAN Workshop on Chinese Language Processing, 2006, 17-24.

[6] 崔世起,刘群,林守勋等.中文缩略语自动抽取初探[C].全国第八届计算语言学联合学术会议(JSCL-2005).

[7] 武子英,郑家恒.现代汉语缩略语自动识别的方法研究[J].计算机工程与设计,2007, 28(16):4052-4054.

[8] Zhifei Li and David Yarowsky. Unsupervised Translation Induction for Chinese Abbreviations using Monolingual Corpora. [C] In Proceedings of ACL-08, 2008, 425–433.

[9] Danushka Bollegala, Yutaka Matsuo and Mitsuru Ishizuka. Extracting key phrases to disambiguate personal name queries in web search. [C] CLIIR '06 Proceedings of the Workshop on How Can Computational Linguistics Improve Information Retrieval, 2006, 17-24.

[10] Vinay Bhat, Tim Oates, Vishal Shanbhag and Charles Nicholas. [J] Finding aliases on the web using latent semantic analysis. Data & Knowledge Engineering, 2004, 49: 129–143.

[11] Winston H. Hsu, Lyndon S. Kennedy and Shih-Fu Chang. Video search reranking through random walk over document-level context graph. [C] MULTIMEDIA '07 Proceedings of the 15th international conference on Multimedia, 2007, 971-980.