文章编号：1003-0077(2013)02-0001-09

# 一种融合实体语义知识的实体集合扩展方法

齐振宇，刘 康，赵 军

（中国科学院 自动化研究所 模式识别国家重点实验室，北京 100190）

**摘 要**：实体集合扩展是开放式信息抽取的一个重要问题，该问题研究如何从一个语义类的若干实体（称为种子）出发，得到该类别的更多实体。现有实体集合扩展方法主要使用上下文模板或种子在语料中的分布信息进行抽取，其缺点是无法解决种子的歧义问题，而该问题会影响方法的有效性。在该文中，作者提出了一种融合实体语义知识的实体集合扩展方法，通过引入语义知识来解决种子歧义性问题。新方法通过使用 Wikipedia 实现了语义知识的引入，并把基于语义知识的扩展方法和基于模板的扩展方法相融合。实验表明，与单纯基于上下文方法相比，该文方法在准确率上提升了 18.5%，召回率上提升了 6.8%，MAP 值上提升了 22.8%。

**关键词**：实体集合扩展；知识库；语义知识

**中图分类号**：TP391 **文献标识码**：A

## A Novel Entity Set Expansion Method Leveraging Entity Semantic Knowledge

QI Zhenyu, LIU Kang, ZHAO Jun

(National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing 100190, China)

**Abstract**: Entity Set Expansion is one of the important problems in Open Information Extraction. Entity Set Expansion refers to expanding several given seeds of one concept into a more complete set. Most approaches solve the problem by using context or distributional information, suffering from the limitation of seed ambiguity problem which results in poor results. In this paper we present a novel method which introduces the semantic knowledge by leveraging Wikipedia knowledge base. We combine this method with traditional template based method. Experiment results show that the proposed method improves 18.5% in precision, 6.8% in recall and 22.8% in MAP.

**Key words**: Entity Set Expansion; knowledge base; semantic knowledge

## 1 引言

无论在学术领域还是在工业领域，同类命名实体列表都具有广泛的应用。在工业领域中，搜索引擎公司比如 Google、Yahoo、Bing 等都在后台维护大量的命名实体列表以提高用户体验[1]；而在学术领域，同类命名实体列表在问答系统、知识库构建等方面也有重要应用[2]。鉴于此，多年来一直有研究者在研究如何获得同类实体列表，即如何解决实体集合扩展问题。

实体集合扩展（Entity Set Expansion），指的是这样一类问题：给定某语义类的若干实体（以下称为“种子”），要求得到该类别的更多实体。比如已知{中国、美国、俄罗斯}三个国家，要求找出更多国家，比如{德国、日本、巴西…}。

目前解决实体集合扩展问题主要有三类方法，分别是基于分布的方法、基于模板的方法以及基于

**收稿日期**：2011-11-22 **定稿日期**：2012-04-05

**基金项目**：国家自然科学基金资助项目（61070106，61202329，61272332）；国家 973 计划资助项目（2012CB316300）；中国科学院先导专项资助项目（XDA06030300）；国家 863 高科技计划资助项目（2012AA011102）；清华大学信息科学与技术国家实验室（筹）资助

**作者简介**：齐振宇（1985—），男，博士研究生，研究方向为自然语言处理和信息检索；刘康（1981—），男，助理研究员，研究方向为自然语言处理和信息检索；赵军（1966—），男，研究员，研究方向为自然语言处理和信息检索。

融合的方法。无论哪一类方法，都需要若干种子作为出发点，进而通过提取种子出现的上下文或者网页结构等信息来捕获种子共有的浅层环境特征，以此作为扩展的依据。但实体往往具有歧义性，这会导致所提取的浅层环境特征产生偏差，使得扩展得到的实体与原始种子在语义上不相匹配，从而影响实体集合扩展的精度。

比如“Washington”作为种子，可能指地点（美国首都）、可能指人物（美国前总统），也可能指一艘船（美国海军的一艘航空母舰）等。不同语义的“Washington”，其上下文具有不同特征。如果不区分各自的语义，简单地把不同语义的上下文特征混杂在一起，这种情况下得到的浅层环境特征是不可靠的，基于此得到的扩展结果也会包含很多错误。

为解决种子歧义性问题，本文提出了一种通过挖掘种子实体蕴含的语义信息，并与基于上下文统计特征的方法结合起来进行实体集合扩展的方法。具体地，我们在 Wikipedia 中利用种子实体的链接信息和类别信息进行实体集合扩展，在此基础上融合了传统的基于浅层环境特征的实体集合扩展方法。实验结果表明，本文方法可以同时提高准确率、召回率和 MAP 值。

本文组织结构如下：第 2 节介绍实体集合扩展的研究现状以及本文的动机；第 3 节介绍 Wikipedia 及其中蕴含的语义信息在实体集合扩展中的应用；第 4 节介绍本文提出的融合实体语义知识的实体集合扩展方法；第 5 节是实验结果与分析；最后给出总结与展望。

## 2 相关工作

实体集合扩展的目标可以分为两类，一类是大而开放类别的语义类，另一类是小而封闭类别的语义类[3]。在前者中，待扩展的语义类含有数量庞大的实体（即“大”），而且其实体可以是变化的（即“开放”）；比如“运动员”这个类别。而在后者中，待扩展的语义类规模较小（即“小”），而且其实体基本没有变化（即“封闭”）；比如“国家”这个类别。本文主要研究针对小而封闭语义类的实体集合扩展问题。

目前解决实体集合扩展问题的主流方法，大体上可以分为基于模板、基于分布以及基于融合等三大类。

基于模板的方法，代表性工作包括文献[3-7]等。这类方法的核心思想是，通过某种方式得到模板，利用模板抽取候选实体，最后对候选进行打分排序得到结果。这里的模板可以是预先定义的语义模板，比如“such as”，“and”等，也可以是种子在语料中出现的高频上下文。实验结果[6]表明这类方法适用于处理小而封闭的语义类扩展问题。

基于分布的方法，代表性工作包括文献[8-10]等。这类方法核心思想是，统计语料库中每个词项的上下文分布并构造词项分布矩阵，利用该矩阵计算每个词项与种子的相似度，以此作为打分和排序的标准。这一类方法更适用于处理大而开放的语义类扩展问题。

基于融合的方法，代表性工作包括文献[11-12]等。这类方法使用多种类型的数据（比如普通网页文本、网页表格、查询日志等），对不同类型的数据采用不同处理方法（基于模板或基于分布），并对各自的结果进行融合。这种方法可以降低单一方法所产生错误对总体结果的影响，这类方法同样更适用于处理大而开放的语义类扩展问题。

已有方法均从种子的上下文统计特征入手，没有使用种子的语义知识。这些方法的弊端在于单纯的上下文统计特征不足以完整刻画种子的全部特性。尤其当种子具有歧义性时，其上下文统计特征会产生偏差，此时扩展效果也会受到很大影响。

比如“Lincoln”这个词，可能指人物，可能指轿车，还可能指地点。而不同语义的“Lincoln”，其上下文统计特征显然具有不同规律。当指人物时，其上下文多为“…是美国总统”，“…出生于”等；而指轿车时，其上下文多为“…报价”，“…是一种豪华车”等；而指地点时，其上下文多为“…创建于”“…位于”等。

为降低种子歧义性对实体集合扩展的不良影响，我们提出一种融合实体语义知识的实体集合扩展方法。该方法把基于语义知识的扩展方法与基于模板的扩展方法融合在一起。在基于语义知识进行扩展时，引入语义知识库来挖掘种子蕴含的语义信息，并利用这些语义信息在知识库中进行扩展，以此降低种子歧义性的影响；在基于模板进行扩展时，使用种子的上下文进行扩展，以此弥补知识库在更新速度和完备性上的不足。最后把这两种方法的结果融合起来作为最终结果。

## 3　Wikipedia 及其所蕴含语义知识的使用

### 3.1　Wikipedia

Wikipedia① 是一个基于 Wiki 技术的百科全书项目，其产物是一个动态的、可自由访问和编辑的知识体。根据知名的 Alexa 网络流量统计排名，Wikipedia 目前为世界网站流量排名第七大网站。

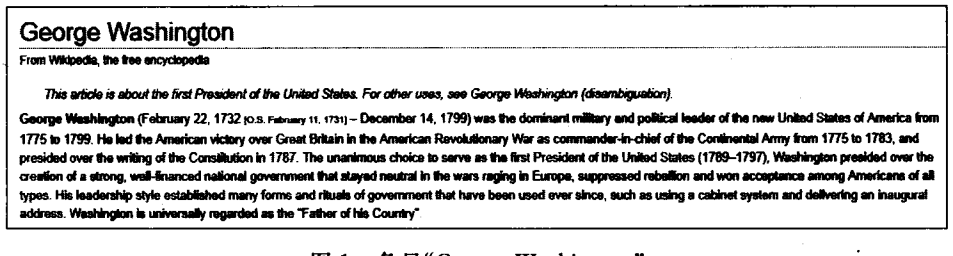
截至 2011 年 11 月，Wikipedia 的总条目已达 1900 万条(其中英文条目超过 370 万条②)，每个条目对应一篇高质量的、富含超链接以及类别信息的文档。Wiki 基金会每个月都会放出一个新版本的 Wikipedia 下载以满足用户的需求。

由于 Wikipedia 具有信息量大、质量高、获取方便、更新快等优点，其在自然语言处理以及知识工程领域得到广泛的使用，大量的研究工作使用 Wikipedia 作为语料或资源。本文使用英文版的 Wikipedia 作为语义知识库。

### 3.2　Wikipedia 所蕴含语义知识在实体集合扩展中的使用

本文利用 Wikipedia 中条目之间的超链接关系以及类别标签体系两类语义知识进行实体集合扩展。

**超链接关系**：在 Wikipedia 中，每个条目对应一篇文档(见图 1)，该文档是对该条目的描述。平均每篇 Wikipedia 文档含有 34 个到其他条目的链接，同时有 34 个其他条目链接到该文档[13]，本文将这些链接视为对该条目的一种语义描述。


George Washington

From Wikipedia, the free encyclopedia

*This article is about the first President of the United States. For other uses, see George Washington (disambiguation).*

**George Washington** (February 22, 1732 [O.S. February 11, 1731] – December 14, 1799) was the dominant military and political leader of the new United States of America from 1775 to 1799. He led the American victory over Great Britain in the American Revolutionary War as commander-in-chief of the Continental Army from 1775 to 1783, and presided over the writing of the Constitution in 1787. The unanimous choice to serve as the first President of the United States (1789–1797), Washington presided over the creation of a strong, well-financed national government that stayed neutral in the wars raging in Europe, suppressed rebellion and won acceptance among Americans of all types. His leadership style established many forms and rituals of government that have been used ever since, such as using a cabinet system and delivering an inaugural address. Washington is universally regarded as the "Father of his Country".

图 1　条目“George Washington”

**类别标签体系**：Wikipedia 中，一个条目可以有若干个类别标签(Category Label，见图 2)，而一个标签可以标注多个条目。这些标签反映了该条目所属的语义类别。此外，所有类别标签组成了一个复杂的类别体系。通过衡量单个标签在分类体系中的位置可以考察不同标签的语义信息，进而考察属于该标签的条目的语义信息。

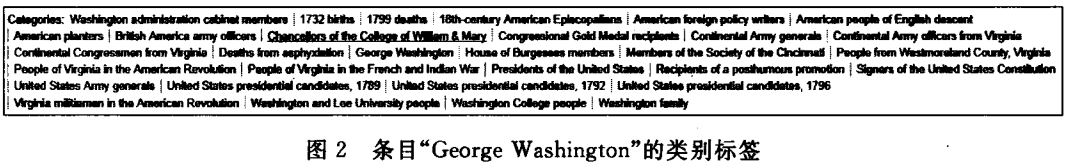

Categories: Washington administration cabinet members | 1732 births | 1799 deaths | 18th-century American Episcopalians | American foreign policy writers | American people of English descent | American planters | British America army officers | Chancellors of the College of William & Mary | Congressional Gold Medal recipients | Continental Army generals | Continental Army officers from Virginia | Continental Congressmen from Virginia | Deaths from asphyxiation | George Washington | House of Burgesses members | Members of the Society of the Cincinnati | People from Westmoreland County, Virginia | People of Virginia in the American Revolution | People of Virginia in the French and Indian War | Presidents of the United States | Recipients of a posthumous promotion | Signers of the United States Constitution | United States Army generals | United States presidential candidates, 1789 | United States presidential candidates, 1792 | United States presidential candidates, 1796 | Virginia militiamen in the American Revolution | Washington and Lee University people | Washington College people | Washington family

图 2　条目“George Washington”的类别标签

在本文中，我们使用条目之间的超链接关系进行消歧以及相关度的计算，使用类别标签体系进行扩展。在下一节中，我们会详细描述这两类语义知识的使用。

## 4　融合实体语义知识的实体集合扩展方法

在这一节中，我们将介绍融合实体语义知识的实体集合扩展方法。其中 4.1 节介绍基于语义知识的扩展方法；4.2 节介绍基于模板的扩展方法；4.3 节介绍二者的融合。

### 4.1　基于语义知识的实体集合扩展方法

首先我们做出一个基本假设：**种子实体在 Wikipedia 中都能找到对应条目**。

实际上，Wikipedia 不可能覆盖所有的实体。但由于人们通常使用熟悉的实体作为种子，而 Wikipedia 对于常见实体的覆盖度非常高。所以这个假设可以成立。

① http://www.wikipedia.org/

② http://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia

结合 Wikipedia 知识库的结构特点，我们设计了一个三阶段的实体集合扩展系统：

第一阶段—消歧阶段。本阶段的任务是明确种子实体在 Wikipedia 中对应的条目。输入是 3 个种子，输出是这 3 个种子在 Wikipedia 中对应的条目。

第二阶段—扩展阶段。本阶段的任务是根据上一阶段的结果，找到 Wikipedia 中可能属于同语义类的条目作为候选。输入是种子对应的条目，输出是作为候选的若干条目。

第三阶段—选取阶段。本阶段的任务是对候选条目打分并排序，得到最终的结果。

以下分别介绍这三个阶段。

4.1.1 消歧阶段

当确认种子实体在 Wikipedia 中对应的条目时，由于种子锚文本可能具有多种不同语义，会出现歧义问题。关于这个问题我们通过一个例子加以说明。

仍然考虑“Washington”这个词，它可以指向很多实体：

**人**：*1. 美国第一任总统华盛顿。2. 其他姓“华盛顿”的人，比如电影演员丹泽尔·华盛顿*

**地点**：*1. 美国首都。2. 美国的一个州名。3. 其他地点。*

**船**：*1. 美国海军华盛顿号航空母舰。*

**机构**：*1. 华盛顿大学。2. 其他机构。*

实际上，“Washington”作为锚文本在 Wikipedia 中指向的条目超过 100 个。这种一个锚文本指向多个条目的现象非常普遍。下一节实验数据表明，每个实体在 Wikipedia 中作为锚文本指向的条目平均超过 5 个。为确认种子实体对应的条目，必须解决歧义问题。

我们提出了一种综合考虑条目概率与语义相关度的消歧方法。设种子 $A$ 对应的候选条目集为 $\{a_1,a_2,\cdots,a_i\}$，种子 $B$ 对应的候选条目集为 $\{b_1,b_2,\cdots,b_j\}$，种子 $C$ 对应的候选条目集为 $\{c_1,c_2,\cdots,c_k\}$，给定 $A$、$B$、$C$ 后，我们按式(1)的方法选取最合适的条目组 $\{a_l,b_m,c_n\}$。

$$\begin{aligned}&\operatorname{argmax}(a_l,b_m,c_n)\\&=\lambda\times P(a_l,b_m,c_n)+(1-\lambda)\times R(a_l,b_m,c_n)\end{aligned}\tag{1}$$

该方法考察两方面的因素：第一是三个候选条目被选中的概率得分；第二是三个候选条目之间的相关度得分。

其中概率得分通过计算三个条目被选中概率之积求对数得到：

$$P(a_l,b_m,c_n)=\ln(Pa_l)+\ln(Pb_m)+\ln(Pc_n)\tag{2}$$

而相关度得分通过求三个条目两两之间相关度得到：

$$\begin{aligned}P(a_l,b_m,c_n)=\ln[&\mathrm{rel}(a_l,b_m)+\mathrm{rel}(b_m,c_n)\\&+\mathrm{rel}(a_l,c_n)]\end{aligned}\tag{3}$$

其中，我们使用 Milne[13] 提出的算法来计算两个条目之间语义相关度，见式(4)。

$$rel(x,y)=1-\frac{\log(\max(|X|,|Y|))-\log(|X\cap Y|)}{\log(|W|)-\log(\min(|X|,|Y|))}\tag{4}$$

其中 $x$、$y$ 是两个条目，$X$、$Y$ 分别是链接到这两个条目的其他条目的集合，$W$ 指整个 Wikipedia。

$\lambda$ 作为参数调节两部分所占的比重。

4.1.2 扩展阶段

我们设计了一种使用 Wikipedia 中类别标签来寻找与种子条目属于同一语义类条目的扩展系统，该系统框架见图 3。该系统可以分为两个阶段：

阶段一，求出 $a$、$b$、$c$ 三个条目中每个条目的标签集合，记为 $La$、$Lb$、$Lc$。取出至少在两个标签集合中出现的标签组成一个公共标签集合。

阶段二，抽取出公共标签集合中标签包括的文章作为候选条目。

为提高抽取效果，我们在阶段一中进行标签扩展：即 $La$ 不仅包括条目 $a$ 的标签，还包括条目 $a$ 的标签在标签体系中的上一层标签。

4.1.3 候选选取阶段

我们考察候选条目与种子条目之间的相关度，并选取相关度在一定阈值(以下称该值为“相关度阈值”)以上的候选条目作为结果。

在训练阶段，为求得“相关度阈值”，我们采取以下做法：设训练集 $T$ 共有 $m$ 个语义类，对其中每个语义类 $s$，我们利用 4.1.1 节中的式(4)计算同类别条目之间的平均相似度：

$$\mathrm{AvgRel}_s=\frac{\sum_{(x\in s,y\in s,x\neq y)}rel(x,y)}{n\times(n-1)/2}\tag{5}$$

其中 $n$ 为类 $s$ 含有的实体个数。之后对 $T$ 中 $m$ 个语义类的平均相似度求平均，得到整个训练集 $T$ 的平均语义相似度：

$$\mathrm{AvgRel}_T=\frac{\sum_{i=1}^{m}\mathrm{AvgRel}_i}{m}\tag{6}$$

在本文中，经计算得到的“相关度阈值”为 $\mathrm{AvgRrel}_T=0.56$。

在测试阶段，我们利用 4.1.1 节中式(4)分别

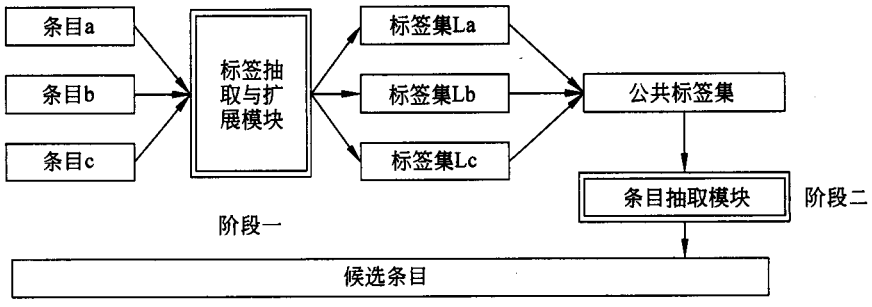



图 3　扩展系统

计算候选条目与三个种子条目之间的相关度并求平均值。抽取平均相关度在“相关度阈值”以上的候选条目组成结果集，并将结果集合中的条目按相关度排序作为结果返回。

### 4.2　基于模板的实体集合扩展

我们实现了一种高效的基于模板的实体集合扩展方法[6]。该算法把三个种子作为查询词送到搜索引擎中，并爬取搜索引擎返回的前 100 个 URL 对应的网页作为语料。之后针对单个网页学习种子在其中出现的模板，并利用学到的模板得到候选。最后我们采用按照出现频率排序的方式为候选排序并抽取出现次数大于 1 的候选作为结果返回。

### 4.3　基于语义知识与基于模板两种方法的融合

以上我们通过两种扩展方法得到两个结果集，基于语义知识的抽取结果集合记为 Rs，其中的候选表现为＜候选，相关度＞；基于模板的抽取结果集合记为 Rp，其中的候选表现为＜候选，出现次数＞。我们按以下方法对这两个集合进行融合：

首先把 Rp 中候选的出现次数按式(7)归一化到[0,1]区间。

$$\text{freq}=\frac{\text{frequency}}{\text{frequency}_{\max}} \tag{7}$$

其中分子为该候选出现的次数，分母为该类别候选中出现次数最多候选的出现次数。之后对每个候选按照下面式(8)对其打分：

$$\text{Score(Candidate)}=\alpha\times\text{freq}+(1-\alpha)\times\text{relatedness} \tag{8}$$

其中 freq 和 relatedness 分别为该候选的频率得分和相关度得分。如果该候选只有其中一个得分，那么另一项得分按 0 计算，α 作为参数调节二者之间的权值。

## 5　实验数据与分析

### 5.1　实验数据

本文使用 Wikipedia20110722 版本作为知识库，另外使用了 WikipediaMiner① 软件工具(1.2.0 版本)处理 Wikipedia 数据。

本文构建了 2 组×6 类别/组共 12 个语义类作为实验数据，人工标定每个语义类所包含的实体。其中第一组作为训练集，第二组作为测试集。相关数据如表 1 和表 2 所示。

表 1　训练集相关信息(6 语义类)

| 类　别 | 类别大小 | Wiki 中含有实体数 | Wiki 覆盖率 | Wiki 平均相关条目 |
|---|---|---|---|---|
| Countries | 202 | 202 | 1.00 | 3.10 |
| Elements | 117 | 117 | 1.00 | 3.14 |
| English Premier Football Clubs | 40 | 19 | 0.48 | 3.79 |
| Italian Regions | 18 | 17 | 0.94 | 2.18 |
| Japanese Prefectures | 44 | 43 | 0.98 | 5.21 |
| Texas Counties | 254 | 237 | 0.93 | 14.72 |
| 均值 | 112.50 | 105.83 | 0.89 | 5.35 |

注：如无特殊说明，本文中均值均为宏平均。

① http://wikipedia-miner.cms.waikato.ac.nz/

表2 测试集相关信息(6语义类)

| 类别 | 类别大小 | Wiki中含有实体数 | Wiki覆盖率 | Wiki平均相关条目 |
|---|---|---|---|---|
| Bottled Water Brands | 67 | 28 | 0.42 | 2.64 |
| Cocktails | 263 | 121 | 0.46 | 3.09 |
| Greek Islands | 280 | 112 | 0.40 | 1.86 |
| Maryland Counties | 24 | 22 | 0.92 | 17.59 |
| Roman Emperors | 138 | 129 | 0.93 | 2.51 |
| Stars | 379 | 161 | 0.42 | 2.39 |
| 均值 | 191.83 | 95.50 | 0.59 | 5.01 |

### 5.2 消歧算法效果验证

本实验验证4.1.1节中消歧算法的有效性,对比不进行消歧与进行消歧两种方法的效果(表3)。进行消歧时,按照第4节所述方法进行实体集合扩展。不进行消歧时,采用如下方法:设种子$A$对应的候选条目集为$\{a_1,a_2,\cdots,a_i\}$,种子$B$对应的候选条目集为$\{b_1,b_2,\cdots,b_j\}$,种子$C$对应的候选条目集为$\{c_1,c_2,\cdots,c_k\}$,抽取每个候选条目集中所有条目的标签以及标签的父标签分别组成标签集合$LA$,$LB$,$LC$,取出至少在其中两个标签集合中出现的标签组成一个公共标签集合,抽取出公共标签集合中标签包括的文章作为候选条目。在计算候选条目的相关度时,把候选条目与种子的所有可能条目的相关度均值作为该候选的相关度。

表3 消歧与不消歧效果比较

| | 不进行消歧 | | | 进行消歧 | | |
|---|---|---|---|---|---|---|
| | P | R | MAP | P | R | MAP |
| Bottled Water Brands | 0 | 0 | 0 | 0.689 4 | 0.064 7 | 0.056 8 |
| Cocktails | 0.008 | 0.018 2 | 0.009 | 0.405 4 | 0.078 3 | 0.034 2 |
| Greek Islands | 0.573 9 | 0.119 3 | 0.091 6 | 0.608 8 | 0.257 1 | 0.198 1 |
| Maryland Counties | 0 | 0 | 0 | 0.852 7 | 0.816 7 | 0.747 4 |
| Roman Emperors | 0.116 9 | 0.065 2 | 0.058 7 | 0.594 7 | 0.397 1 | 0.283 5 |
| Stars | 0.02 | 0.006 | 0.004 8 | 0.591 9 | 0.099 7 | 0.079 9 |
| 均值 | 0.119 8 | 0.034 8 | 0.027 3 | 0.623 8 | 0.285 6 | 0.233 3 |

可以看出,由于不进行消歧,候选的相关度偏低,很难达到"相似度阈值",使得结果很差。而进行消歧可以消除歧义,极大地提升方法的表现。

### 5.3 消歧算法中参数λ的确定

本实验确定4.1.1节式(1)中参数λ的值。对每个语义类做100组实验,每组实验随机抽取该类别的3个实体作为种子,记录λ取不同值时3个种子对应的条目,当3个条目都正确时视为正确,否则视为错误。实验结果如表4所示。

表4 消歧阶段λ的确定

| λ取值 | 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Countries | 0.35 | 0.97 | 0.99 | 0.98 | 0.98 | 0.98 | 0.96 | 0.98 | 0.98 | 0.95 | 0.96 |
| Elements | 0.95 | 1.00 | 0.99 | 0.96 | 0.95 | 0.98 | 0.91 | 0.91 | 0.91 | 0.97 | 0.93 |
| English Premier Football Clubs | 0.77 | 0.82 | 0.82 | 0.90 | 0.94 | 1.00 | 1.00 | 0.88 | 0.81 | 0.85 | 0.79 |

续表

| λ取值 | 0.0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Italian Regions | 0.83 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.95 | 0.89 | 0.78 | 0.90 | 0.82 |
| Japanese Prefectures | 0.37 | 0.42 | 0.27 | 0.30 | 0.26 | 0.16 | 0.18 | 0.20 | 0.16 | 0.16 | 0.08 |
| Texas Counties | 0.94 | 0.81 | 0.61 | 0.29 | 0.19 | 0.10 | 0.08 | 0.03 | 0.04 | 0.05 | 0.03 |
| 均值 | 0.70 | 0.84 | 0.78 | 0.74 | 0.72 | 0.70 | 0.68 | 0.65 | 0.61 | 0.65 | 0.60 |

### 5.4　对标签进行扩展的重要性分析

本实验验证 4.1.2 节中标签扩展的重要性(表 5)。对每个语义类做 500 组实验，每组实验随机 3 个种子，依据 4.2 节中描述的算法确认其在知识库中对应条目后，对得到的 3 个条目抽取其标签。可以看出，不进行标签扩展时，平均每组种子只能抽出 2.8 个标签；进行扩展后每组种子可以抽出 10.3 个标签。这可以证明加入标签扩展后，大大增加了得到相关条目的可能。

**表 5　扩展阶段进行标签扩展的重要性分析**

| 类　别 | 进行标签扩展 | | 不进行标签扩展 | |
|---|---|---|---|---|
| | 3 条目公共标签个数 | 2 条目公共标签个数 | 3 条目公共标签个数 | 2 条目公共标签个数 |
| Countries | 6.54 | 12.38 | 1.14 | 3.41 |
| Elements | 5.56 | 2.54 | 1.07 | 1.00 |
| English Premier Football Clubs | 9.86 | 5.12 | 2.30 | 2.24 |
| Italian Regions | 8.35 | 3.22 | 2.13 | 1.03 |
| Japanese Prefectures | 2.40 | 2.57 | 0.47 | 0.65 |
| Texas Counties | 2.69 | 0.92 | 0.86 | 0.45 |
| 均值 | 5.90 | 4.46 | 1.33 | 1.46 |

### 5.5　候选选取过程中相关度阈值影响

本实验验证 4.1.3 节中相关度阈值对抽取条目的影响。我们对每个类别进行 50 组实验，每组实验随机选取 3 个种子，通过第 4 节介绍的算法在 Wikipedia 中进行扩展。加入相关度阈值和不加入两种情况下的实验结果如表 6 所示。

**表 6　按照相关度规则抽取条目**

| | 候选条目 | 正确候选 | 相关度阈值内候选 | 相关度阈值内正确候选 | 正确候选平均相关度 | 错误候选平均相关度 |
|---|---|---|---|---|---|---|
| Countries | 601.16 | 178.80 | 22.35 | 19.20 | 0.43 | 0.10 |
| Elements | 365.67 | 117.00 | 142.02 | 102.08 | 0.65 | 0.20 |
| English Premier Football Clubs | 1 355.94 | 18.02 | 120.14 | 18.00 | 0.74 | 0.19 |
| Italian Regions | 583.82 | 16.00 | 23.61 | 14.20 | 0.64 | 0.11 |
| Japanese Prefectures | 245.29 | 22.16 | 44.71 | 19.45 | 0.61 | 0.14 |
| Texas Counties | 361.12 | 229.24 | 257.29 | 220.41 | 0.73 | 0.24 |
| 均值 | 585.50 | 96.87 | 101.69 | 65.56 | 0.63 | 0.16 |

从上表可以得到以下两个结论：

1. 正确候选条目和错误候选条目，二者与种子的相关度有很大差别(0.63 Vs 0.16)，引入相关度作为衡量候选准确性指标非常合理。

2. 引入相关度阈值后，在损失1/3正确候选条目(96 Vs 65)的前提下，滤去了4.7倍的错误候选条目(585 Vs 101)。这证明了引入相关度阈值的有效性。

### 5.6 融合语义知识与统计信息的实体集合扩展

本实验对比单纯基于模板、单纯基于语义以及二者融合三种方法的实体集合扩展结果。我们对每个类别做5组实验，每组随机抽取3个种子，分别使用基于模板、基于语义以及二者融合三种方法进行实体集合扩展，最后对5组实验的结果求均值。实验结果如表7所示。

表7 三种方法的结果比较

| 类别 | 单纯基于模板 | | | 单纯基于语义 | | | 二者融合 | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | MAP | P | R | MAP | P | R | MAP |
| Bottled Water Brands | 0.297 7 | 0.585 1 | 0.374 1 | 0.689 4 | 0.064 7 | 0.056 8 | 0.303 7 | 0.594 0 | 0.481 9 |
| Cocktails | 0.090 8 | 0.638 0 | 0.350 2 | 0.405 4 | 0.078 3 | 0.034 2 | 0.112 3 | 0.660 8 | 0.369 0 |
| Greek Islands | 0.165 7 | 0.935 0 | 0.571 2 | 0.608 8 | 0.257 1 | 0.198 1 | 0.165 8 | 0.937 1 | 0.668 2 |
| Maryland Counties | 0.228 2 | 0.675 0 | 0.564 6 | 0.852 7 | 0.816 7 | 0.747 4 | 0.414 5 | 0.933 3 | 0.918 7 |
| Roman Emperors | 0.075 2 | 0.873 9 | 0.516 4 | 0.594 7 | 0.397 1 | 0.283 5 | 0.076 4 | 0.892 8 | 0.655 9 |
| Stars | 0.303 7 | 0.853 8 | 0.793 9 | 0.591 9 | 0.099 7 | 0.079 9 | 0.304 3 | 0.855 9 | 0.799 8 |
| 均值 | 0.193 6 | 0.760 1 | 0.528 4 | 0.623 8 | 0.285 6 | 0.233 3 | 0.229 5 | 0.812 3 | 0.648 9 |

可以看出，传统单纯基于模板的方法召回率较高，但由于受到种子歧义性问题的影响，准确率较低；而单纯基于语义的方法解决了种子歧义性问题，所以准确率较高，但召回率较低。二者融合以后，P值提升了18.5%，R值提升了6.8%，而MAP也提升了22.8%，这说明二者融合的方法吸收了两种方法各自的优点，弥补了不足，使得整体结果有了很大的提升。

另外，我们也测试了式(8)中不同α对融合结果的影响，可以看出，融合语义知识后的效果比单纯使用模板的效果要好，而当α取0.2时，融合结果最好。

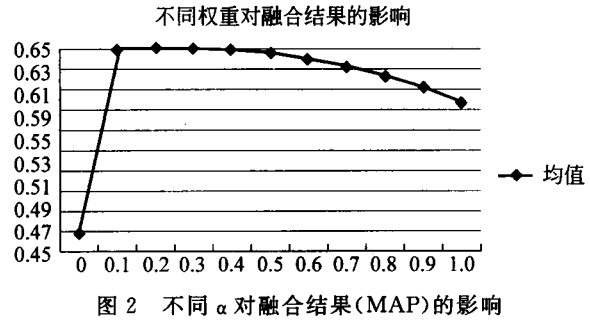



图2 不同α对融合结果(MAP)的影响

## 6 总结与展望

实体集合扩展问题是开放式信息抽取中一个重要问题。目前解决该问题的方法基本都是从若干个种子出发，利用模板或种子的分布信息进行扩展，没有考虑到种子的语义信息，所以无法解决种子歧义性问题。

本文提出了使用种子的语义信息进行扩展以解决种子歧义性问题的思路，并利用Wikipedia作为语义知识库，实现一种基于语义知识的扩展方法。在给定种子的情况下，经过消歧阶段、扩展阶段、选取阶段最终得到扩展结果。

此外，本文把基于语义知识的扩展和基于模板的扩展相融合。实验结果表明，新方法在P值上提升了18.5%，R值上提升了6.8%，MAP值提升了22.8%，这证明了本文方法的有效性。

未来工作主要包括以下几个方向：

1. 提升消歧阶段准确率。消歧阶段对后续工作影响重大，接下来我们考虑引入更丰富的语义知识，比如类别标签等来提升消歧阶段的准确率。

2. 采用更好的融合方式。目前我们使用的是简单的线性融合，以后还可以探索其他融合方式。

## 参考文献

[1] Vishnu Vyas, Patrick Pantel, Eric Crestan. Helping editors choose better seed sets for entity set[C]//Proceedings of CIKM 2009. Hong Kong: ACM, 2009:

225-234.

[2] Richard C Wang, Nico Schlaefer, William W Cohen et al. Automatic Set Expansion for List Question Answering[C]//Proceedings of EMNLP 2008. USA: ACL, 2008: 947-954.

[3] Richard C Wang, William W Cohen. Automatic Set Instance Extraction Using the Web [C]//Proceedings of ACL/AFNLP 2009. Singapre: ACL, 2009: 441-449.

[4] Luis Sarmento, Valentiin Jijkoun. "More Like These": Growing Entity Classes from Seeds [C]// Proceedings of CIKM 2007. Portugal: ACM, 2007: 959-962.

[5] Pasca. Weakly-supervised discovery of named entities using web search queries[C]//Proceedings of CIKM 2007. Portugal: ACM, 2007: 683-690.

[6] Richard C Wang, William W Cohen. Language-Independent Set Expansion of Named Entities Using the Web[C]//Proceedings of ICDM 2007. USA: IEEE Computer Society, 2007: 342-350.

[7] Richard C Wang, William W Cohen. Iterative set expansion of named entities Using the web[C]//Proceedings of ICDM 2008. Italy: IEEE Computer Society, 2008: 1091-1096.

[8] Patrick Pantel, Eric Crestan, Arkady Borkovsky, et al. Web-Scale Distributional Similarity and Entity Set Expansion[C]//Proceedings of EMNLP2009. Singapore: ACL, 2009: 938-947.

[9] Benjamin Van Durme, Marius Pasca. Finding Cars, Goddesses and Enzymes Parametrizable Acquisition of Labeled[C]//Proceedings of AAAI08. USA: AAAI Press 2008: 1243-1248.

[10] Yeye He, Dong Xin. SEISA Set Expansion by Iterative Similarity Aggregation [C]//Proceedings of WWW 2011. India: ACM, 2011:427-436.

[11] Partha Pratim Talukdar, Joseph Reisinger, et al. Weakly-supervised acquisition of labeled class instances using graph random walks[C]//Proceedings of EMNLP 2008. USA : ACL, 2008 :582-590.

[12] Marco Pennacchiotti, Patrick Pantel. Entity Extraction via Ensemble Semantics [C]//Proceedings of EMNLP 2009. Singapore: ACL, 2009: 238-247.

[13] David Milne, Ian H Witten. Learning to link with Wikipedia [C]//Proceedings of CIKM 2008. USA: ACM, 2008 :509-518.

# 中国中文信息学会常务理事王海峰博士出任 ACL 主席

中国中文信息学会常务理事王海峰博士出任自然语言处理领域最具影响力的国际学术组织"计算语言学协会"(Association for Computational Linguistics,简称 ACL)主席(President)。他也是该协会成立 50 年以来的首位华人主席。

王海峰博士现任百度基础技术领域首席科学家,曾任微软中国研究院副研究员、isilk. com 研究科学家(香港特区政府优秀人才计划),东芝(中国)研究开发中心副所长兼研究部部长、首席研究员等。主要研发领域包括搜索技术、自然语言处理、机器翻译、推荐与个性化、语音及多媒体技术等。主持及参与了多个产品的研发,并申请中国、美国及日本专利 30 余项。除产品应用外,他主持开发的系统在国际机器翻译公开评测中获得多项第一。在政府和科研界,王海峰有很高的认可度,目前兼任北京大学语言信息工程系系主任、哈尔滨工业大学博士生导师、中国中文信息学会常务理事、全国信息技术标准化技术委员会委员等,并在 IJCAI、ACL、SIGIR、KDD、COLING 等多个顶级国际会议中任各类主席职位。鉴于他在科技领域的卓越表现,王海峰被评为"2011 年度中关村高端领军人才"。

中国中文信息学会全体同仁谨在此对王海峰博士出任 ACL 主席表示热烈的祝贺,并衷心祝愿他在任期期间,为在国际范围内推动自然语言处理领域的发展,同时推动中文信息处理事业的发展,做出新的贡献!