文章编号：1003-0077(2012)01-0042-09

基于同义实体扩展的冗余信息去重

姜孟晋，周雅倩，黄萱菁

（复旦大学 计算机科学技术学院，上海 201203）

摘 要：冗余信息去重是信息抽取中的重要任务，对于多元素表示的信息，该文针对以往对各个元素统一处理所存在的问题，将信息元素进行分类，由各类元素的冗余判断难易出发，归纳相似度计算方法，并将各相似度作为特征，通过分类器判断信息间的冗余性。同时对最难判断的命名实体信息元素，该文从其他易判断相似性的信息元素出发，通过同义命名实体的自动扩展，提高信息去重的效果。

关键词：信息抽取；信息去重；命名实体

中图分类号：TP391 **文献标识码**：A

Synonymous Entity Expansion Based Information De-duplication

JIANG Mengjin, ZHOU Yaqian, HUANG Xuanjing

(School of Computer Science, Fudan University, Shanghai 201203, China)

Abstract: Information De-duplication is an important task of Information Extraction. This paper focuses on the multi-field information de-duplication. Previous works usually treat each information field equally. We separate information fields into several categories, generalize the computing method of similarity for each single filed, and use those similarities as the features in a machine learning method to distinguish duplicate information pairs. For the most difficult named entity field, we expand co-reference pairs by using the other easy predicted fields, and use the expanded knowledge to improve the de-duplication performance.

Key words: information extraction; information de-duplication; named entity

1 引言

对于数量呈爆炸式增长的网页，以及其中包含的海量信息，用户在搜索时，不仅想获得所关注的网页，更想直接拿到需求的结果（例如，用户搜索书籍名，就是想直接获得书籍介绍、评价、网上书店链接等信息）。因此，针对各种不同应用的垂直搜索引擎应运而生。

然而，网络上的信息冗余问题十分严重，而垂直搜索引擎不希望将冗余信息返回给用户，因而要对抽取的信息进行合并。网页信息的冗余，不仅存在于不同的网站之间，有时也存在于同一个网站之中。这些冗余不仅包括网络中恶意的抄袭拷贝，也包括不同信息源之间对同一事物的不同描述。因而，信息冗余判断和去重是一个十分棘手的问题。

信息通常会以若干个元素组成，在抽取信息时会把不同的信息元素区分抽取出，然后将整个信息表示为多个信息元素组成的形式，以事件信息为例，简单的包括了事件名，事件发生的时间和地点等，可表示为：事件信息＝{事件名＋发生时间＋发生地点}。

信息在这种多元表示下，去重任务就要以每个信息元素的相似性为基础，从以往的研究来看，命名实体信息元素的冗余判断往往是信息去重中最难处理的一环。

例如，在处理表 1 的五个信息时，1，2，3 事件虽然地名不同，但是事件名和时间非常相似，而实际上

收稿日期：2011-01-15 **定稿日期**：2011-04-28

作者简介：姜孟晋（1985—），男，硕士研究生，主要研究方向为自然语言处理；周雅倩（1976—），女，讲师，主要研究方向为自然语言处理；黄萱菁（1973—），女，教授，研究方向为自然语言处理。

“上海大舞台”和“上海体育馆”是同一地名，如果没有非常完备的同义外部知识库支持，很难仅从文本上得到两个地名实体是同义的。一种简单的想法是利用事件名和时间这两个信息元素的相似性，自动扩展识别出地名元素的相似性，从而可以再利用扩展的知识去解决复杂的去重问题。事件4和5，在事件名上差异很大，但是如果前面能学习得到两个记录的地名的一致，就能很好地进行冗余判断。

表1　冗余信息范例

事件标号	事件名	时间	地点
1	刘若英2010年上海演唱会	2010-05-08	上海大舞台
2	2010刘若英上海演唱会	2010-05-08	上海体育馆
3	刘若英演唱会	2010-05-08	上海大舞台
4	声动2010 Singing on Pandora张韶涵潘多拉星球巡回演唱会	2010-05-22	上海体育馆
5	张韶涵上海演唱会	2010-05-22	上海大舞台

文献[1]提出了一种可学习的文本距离函数来计算信息元素间的相似度。它从信息元素的相似度出发，设计若干个相似度函数，应用到每一个信息元素上，并以此为特征，通过机器学习的办法进行分类，最后给出了SVM分类的结果，对信息元素进行去重。但文献[1]对不同类型的信息元素使用了同样的相似度计算函数，实际情况是，不同的信息元素在相似度判断上有简有难，对于格式化的信息，或者枚举型的信息，应用这些相似度函数，不仅复杂化，而且没能利用到信息元素格式上的特点，效果反而不好；而命名实体信息仅靠文本相似度来判断又不能很好解决。

如果对这种多元素信息的每个元素有个简单分类，然后分别给以不同的相似性函数，可以把它们的相似性判断独立出来，以解决同一个相似度函数带来的问题。本文把这种信息元素分为四个大类，枚举元素，格式化元素，命名实体元素和自由文本元素，涵盖了所有可能的信息元素，对不同类型元素可以应用复杂程度不同的相似度函数。

另外，前述也特别提到了，对于命名实体信息元素，在不利用外部知识的情况下，仅从得到的信息集出发，本文从其他简单的信息元素出发，通过求两个命名实体所在信息子集的覆盖程度，自动扩展出一些同义实体词对，并将这些同义实体词对知识加入冗余信息去重过程中，从而提高了信息去重的效果。关于信息去重和同义实体扩展的介绍将放在文章第2节。

2　信息的表示和去重

信息抽取是指将文档中的有用内容进行结构化处理，以表格等形式进行组织，方便存储、检索等进一步应用。信息抽取一般在自由文本上进行，对于网页信息的抽取，可以利用网页的半结构化特点辅助抽取。

抽取得到的信息通常都可以用元组的形式来表示，每条信息可以用一个多元组来表示：

$$T_i = \{t_i^1, t_i^2, \cdots, t_i^n\}$$

元组中的元素对应该信息的某个特定属性。例如，一个书籍信息可以用{书名，作者，出版社，价格}，这样的四元组来表示；而一个学生的信息可以用{姓名，性别，年级}这样的三元组来表示。

2.1　信息元素分类

信息元素可以依照多种方式分类。本文为了比较好地对抽取的信息进行冗余判断，将信息元素按不同的格式分为四类：枚举元素，格式化元素，命名实体元素和自由文本元素。

枚举元素

许多信息抽取系统都有枚举型的元素，例如，性别、生肖、星座、血性等。由于枚举类信息仅包含有限的取值，因而这类信息在判断是否相似时十分容易，只需看比较字串是否相同即可。

格式化元素

信息抽取系统在处理某些特定信息时，由于存储或应用的需要，往往通过规则将其转化为特定格式，例如，时间、邮箱、电话号码等。因为可以根据格式的具体定义来精确地知道这个信息元素的含义，这类格式化信息在判断是否相似时也比较容易。

命名实体元素

一些信息元素的值是命名实体，例如，人名、地名、组织机构名等，由于命名实体存在“一名多义，多名同义”的问题，不能单从字面上加以区分，一般需要借助到外部知识库。信息抽取系统在这类信息上主要是基于两个命名实体是否同义来判断相似。

自由文本元素

除了以上三种类型信息，其他的都可划分为自由文本信息，多是具体内容，这类信息的相似比较就

类似于字符串之间的相似性比较，当然具体的信息还是可以有一些特定的性质的。

2.2 信息去重流程

信息去重，就是把抽到的表示相同信息的元组合并到一起。若初始的信息集合表示为：

$$T = \{T_i \mid i = 1, \cdots, m\}$$

经过合并后的信息集合表示为：

$$T = U_1 \cup U_2 \cup \cdots \cup U_S$$

其中，要求集合 U_i 内所有的元组信息实际表示同一信息，且 $U_i \cap U_j = \varnothing, \forall i, j$。

这样信息去重任务转化为一个聚类任务，而聚类不可避免地要涉及样本间相似度的计算。这里我们采用层次聚类的方法：首先计算两两样本间的相似度，然后将相似度较高的聚合为一类，自下而上得到整个信息集合的聚类。我们使用一个二类的分类器（判断两个信息元组是否重复）来处理相似度计算的任务。但是，对于数据量非常大的信息数据集来说，两两判断将要处理所有信息对，这平方级别的复杂度，通常是不被允许的，也会大大降低效率。这里我们采用分块（Blocking）的办法[2-3]，来快速选取信息数据集中可能的重复信息对。这主要是通过对信息数据集的初步观察，可以得到类似 Hash 函数的简单判别函数来快速筛选可能的重复信息对。

整个信息抽取和去重的工作可以由下面的流程图（图 1）来表示。

1）从信息源（网页，文档等）中抽取信息，得到信息的元组表示。抽取过程中，对于特定信息元素，可能有一些规范化的工作；

2）对抽取到的信息集进行分块，通过简单的判断条件，Hash 函数等将可能出现重复的信息对找出；

3）两两信息对经过一个判断重复与否的分类器（二类线性分类器，SVM 等），得出重复的信息对；

4）将重复信息对聚类（层次聚类），合并重复的信息对，更新到抽取到的信息数据。

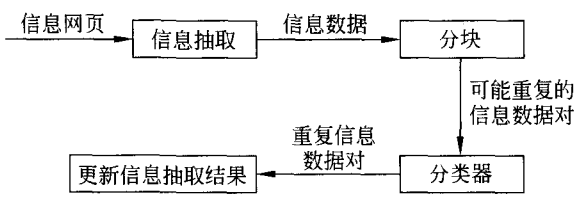

图 1 信息抽取和去重的一般流程

2.3 信息相似度计算

根据信息的元组表示 $T_i = \{t_{1i}, t_i^2, \cdots t_i^n\}$，特别在我们知道信息中各个元素的含义及格式时，我们可以经验性地在信息的特定元素上定义相似度函数 $f^k(t_i^k, t_j^k)$。f^k 仅和第 k 维元素相关，从而两条信息间的相似性可以同过每个元素上相似性的综合得到：

$$\mathrm{sim}(T_i, T_j) = F(f^1(t_i^1, t_j^1), f^2(t_i^2, t_j^2), \cdots, f^n(t_i^n, t_j^n))$$

这里，特征 f^k 的取值范围与元素的类型相关，有的特征是布尔值{0,1}，也有的特征属于区间[0,1]。对于每一维信息元素，可以应用多个相似度函数作为特征。

由先前我们对信息元素的分类，枚举元素和格式化信息元素都可以用简单的匹配函数来区分，或者依据格式设计一些相似度计算函数。命名实体信息元素需要一个同义的词库才能解决，文献[4]基于知网给出了词语间相似度的计算函数，可以用于计算命名实体信息元素之间的相似度，本文 2.5 节还将介绍一种 NE 元素的同义词对自动扩展方法，来提高冗余判断的效果。自由文本信息的相似度问题，作为字串或者句子的相似度，在国内外都已经有过很多研究，文献[5-6]给出了判断中文句子相似度的方法，可以用于计算中文自由文本信息元素之间的相似度，但本文并不仅限中文信息去重问题。文献[7]针对信息元素的匹配问题，对多种不同的字符串比较方法进行实验，本文借鉴之中的方法，根据实际应用环境，在实验中选择合适的字符串相似度计算方法。例如，自由文本信息元素，既可以从向量空间模型上计算相似度，也可以通过计算字符串编辑距离（Levenshtein Distance）作为相似度。

2.4 信息冗余判断

当每个 f^k 都返回布尔值时，F 可使用合取式，否则 F 函数的设计可有很多，当 f^k 返回[0,1]区间上的相似度时，F 可取 f^k 的连乘，$F = \prod_{k=1}^{n} f^k$；或者取各个相似度函数的带权重之和，$F = w_0 + \sum_{k=1}^{n} w_k \times f^k$，这里 w_k 代表第 k 个相似度函数的权重，权重可以由训练样本学习得到。

假定有已经做好分类的信息集，在这基础上可以得到两两信息是否冗余的训练集 $D = \{d_{ij} \mid i \neq j\}$，其中 $d_{ij} = (\{f_{ij}^k \mid k = 1, 2, \cdots, n\}, s_{ij})$，其中 f_{ij}^k 是信息

t_i 和 t_j 利用第 k 个函数计算得到的相似度，s_{ij} 表示信息冗余与否：

$$s_{ij}=\begin{cases}1, & T_i,T_j \text{ 属冗余信息}\\ -1 & T_i,T_j \text{ 不属冗余信息}\end{cases}$$

在有了标注过的一部分语料后，一种简单方法是优化平方误差来训练权重向量 $\vec{w}$：

$$\min_{\vec{w}} E(\vec{w})=\frac{1}{2}\sum_{d\in D}(F_d(\vec{w})-s_d)^2$$

其中 $F_d(\vec{w})=w_0+\sum_{k=1}^{n}w_k * f^k, k=1,2,\cdots,n$，所以 $\vec{w}$ 是 n+1 维的向量。

我们采用梯度下降(Gradient Descent)方法来求得 $\vec{w}$，梯度下降法每次迭代都会选择目标函数梯度最小的方向前进一个步长 $\Delta\vec{w}=-\gamma\nabla E(\vec{w})$。这里 γ 控制了每次的步长，通常称为学习速率，负号是我们想让权重向量向 $E(\vec{w})$ 下降的方向移动。

权重向量的迭代式可写为 $\vec{w}=\vec{w}+\Delta\vec{w}$。迭代的终止条件可以是 $E(\vec{w})$ 的改变小于某个阈值 ε，即，$|E(\vec{w}_i)-E(\vec{w}_{i+1})|<\varepsilon$。由于结果很可能收敛于局部最优解，因此我们将多次随机给定初始权重，从多个最后结果中选择最小值，作为最优解。最后，利用梯度下降方法得到 $\vec{w}$，去计算两条信息之间是否冗余。若 $F_{ij}(\vec{w})>0$，则 T_iT_j 同类，或说冗余，合并做为一类，否则不冗余，全部计算后通过传递闭包关系，可以得到整个信息集的聚类结果。

尽管线性分类的方法已能处理该问题，为了更好的分类结果，我们用核函数方法，将样本映射到高维空间来处理。这里，我们参照文献[1]中的方法，将 T_i,T_j 之间的所有相似度 f^k 作为特征，冗余的信息对组成正例，非冗余的信息对组成负例，使用 SVM 方法进行两两分类，实验中，我们使用 LibSVM 为工具，核函数选择径向基核函数(RBF Kernel)，可以计算得两两信息对之间是否冗余，最后再利用两两信息冗余关系，通过传递闭包关系，合并得到聚类结果。

实验中，我们首先使用经验的方法，确定各个信息元素上的相似度函数，然后用各个相似度的带权重和，来进行线性分类及合并；其次再使用 SVM 分类来确定两两信息样本的相似度，以上的特征为两两信息样本的各种相似度。具体实验方法和相似度函数选择等会在实验部分做具体介绍。

2.5　命名实体信息元素的自动扩展

对于信息记录中的命名实体(Named Entity，NE)元素，我们通常需要指代消解，才能区分出其中互指的元素。在简单的应用中，我们可以维护一个 NE 词表，包扩同义的 NE 词对，这样做去重时，查询这个表格即可判断两个 NE 是否指同一内容。

NE 词表往往需要手工构建，也会借助一些外部的知识库，但这样的词表不能完全适应信息去重任务需要。当我们观察一组重复的信息数据时，往往能够发现，其实对大多数两两重复的信息对，它们在大部分信息元素上的内容是非常相似的，只是有些许信息元素有差别，分歧基本出现在少数的信息元素上。这里，在信息去重的过程中，我们可以借助两条信息记录在其他元素上的相似性，来判断两者在某特定 NE 元素上的相似性，并以此自动扩展一些 NE 同义词对。

假定通过抽取，我们得到了信息集合 $\{T_i|i=1,2,\cdots,m\}$，以及合适的相似性判断函数 $\mathrm{sim}(T_i,T_j)$，每个信息都包含若干信息元素 $T_i=\{t_i^1,t_i^2,\cdots,t_i^n\}$，由于信息的元组表示，元素之间顺序并无关系，这里不妨假设第一维元素 t_i^1 是 NE 元素。

在信息集合 $\{T_i\}$ 上，发现第一维 NE 元素上有两个不同值 a 和 b，且在 NE 词表中 a 和 b 不同义。则对值 a，通过查 NE 词表可以得到所有与 a 同义的 NE(包括 a)，记为 a^*，并通过 a^* 可以找到所有第一维元素被 a^* 包含的信息子集 $T^a=\{T_i^a\}$；同样可以得到 $T^b=\{T_i^b\}$。

由此，我们可以通过相似性判断函数 $\mathrm{sim}(T_i,T_j)$ 计算 T^a 和 T^b 之间的重复对的数目(overlap pairs)，$\mathrm{op}(T^a,T^b)$。(注 1：在此时计算相似度应当忽略在第一维，也就是 NE 元素；注 2：重复对不能重复计算，也即任意条信息记录不能出现在两个重复对中。)

为保证在 $\mathrm{op}(T^a,T^b)$ 的重复对中，来自 T^a 和 T^b 的信息记录都只出现一次，寻找重复对问题，可以转化为二分图的最大匹配问题来解决。将 T^a 和 T^b 中的所有信息记录都看作二分图的顶点，选定相似度阈值 α，当信息间相似度 $\mathrm{sim}(T_i,T_j)>\alpha$ 时，设定 T_i,T_j 之间有边相连，这里 $T_i\in T^a, T_j\in T^b$。这样二分图的最大边匹配数就是原问题的重复对 $\mathrm{op}(T^a,T^b)$ 数目。该问题可在 $O(|T^a||T^b|)$ 时间内，用不断寻找增广路的匈牙利算法来解决。

实际中，我们用下面这个函数来计算两个 NE 元素，a 和 b 的相似度：

$$\mathrm{simNE}(a,b)=\frac{op(T^a,T^b)}{\min(|T^a||T^b|)}$$

使用上述函数对信息集合$\{T_i\}$任意NE词对，都能来计算相似度。实际应用时，需要设定一个阈值t，如果$\text{simNE}(a,b)\geqslant t$，则将$b$设为与$a$同义的NE，并同时在NE词表中，将$b$所关联的NE词$b^*$都加入到$a^*$。

算法1：同义词对自动扩展算法

输入：信息集合$\{T_i\}$，待扩充NE信息元素的对应词表D，阈值t

输出：NE信息元素扩充后的词表D

1：预计算信息间相似度$\text{sim}(T_i,T_j)$，$i\neq j$；

2：对词表中不同义的NE，$\forall a,b\in \text{D},a\neq b$；

3：查找词表中与a,b各自同义的NE集合a^*和b^*；

4：分别获得与a^*和b^*相关的信息子集T^a和T^b；

5：转化为二分图最大匹配问题计算重复对数目：$\text{op}(T^a,T^b)$；

6：如果$\text{simNE}(a,b)=\dfrac{op(T^a,T^b)}{\min(|T^a|,|T^b|)}>t$，且$\text{op}(T^a,T^b)\geqslant 2$，则$a$和$b$同义，更新NE词表；

7：返回更新后的词表；

上述同义词对的自动扩展方法可以继续重复进行，直到没有大于阈值t的词对出现。另外，当信息的元组中不止一个信息元素时，在对某一NE的同义词对进行扩展时，可以先固定其他维NE元素。

实验中，参数选择对结果影响不大，这里并没有过多的参数的调整，当$\text{sim}(T_i,T_j)\geqslant 0.6$时，$T_i$和$T_j$记为重复的信息。当$\text{op}(T_a,T_b)\geqslant 2$且$\text{simNE}(a,b)\geqslant 0.4$时，$a$和$b$才记为同义词对。$\text{op}(T_a,T_b)\geqslant 2$，即重复对数目至少为2，考虑到重复对为1时置信度不足。

3 实验

冗余信息去重实验在信息抽取后的信息集上进行。去重作为信息抽取工作的重要部分，其任务就是在已有的信息集中找到冗余的信息对，进行合并。这里我们分别在两个不同任务上进行了实验。首先我们选择了一个事件信息抽取系统，任务是找到系统抽取的事件中的冗余关系来去重。实验中，我们选择了一部分抽取到的事件，人工标注出其中的冗余关系，作为实验数据集。同时，我们选择了论文索引去重的任务，使用Cora数据集[8]，论文索引由于写法不一致，详略的不一致导致在论文引用分析，索引的合并时产生错误，这里旨在分析出Cora数据集中重复的论文引用，从而去除重复。

实验中一共采用三种方法作为评测的指标：Pairwise，MUC和B-cubed(B^3)。

Pairwise是最直观的评测指标，在信息全集T中，考虑所有可能的信息对，计算准确度和召回率：记实验得到的重复信息对(Duplicate Pair)集合DP_e，标准答案中的信息对集合DP_k，可能的信息对集合至多为$\frac{1}{2}|\text{T}|(|\text{T}|-1)$，则：

$$Precision_{pairwise}=\frac{|\text{DP}_e\cap\text{DP}_k|}{|DP_e|}$$

$$Recall_{pairwise}=\frac{|\text{DP}_e\cap\text{DP}_k|}{|\text{DP}_k|}$$

Pairwise方法只评测了分类器的结果，并未对重复信息聚类结果做评测，因而实验引入MUC和B-cubed这两种共指(co-reference)任务的评测指标，更能体现我们去重任务的要求。

MUC由文献[9]提出，将类内所有冗余的信息看作一个冗余的信息链，并基于冗余信息链做评测。假设标准的去冗余后信息集为$P_k=\{\bigcup_{i=1}^{m}k_i\,|\,i=1,2\ldots,n\}$，实验结果集为$P_e=\{\bigcup_{i=1}^{m}E_i\,|\,i=1,2\ldots,n\}$，计算$P_k$在$P_e$上的投影，可得到每个$K_i$被分为$\pi(K_i)$个集合，则：

$$Pairwise_{\text{MUC}}=\frac{\sum_{i=1}^{n}(|K_i|-|\pi(K_i)|)}{\sum_{i=1}^{n}(|K_i|-1)}$$

交换P_k和P_e在上式中的角色可以计算出$Recall_{MUC}$。

B-cubed由文献[10]提出，在MUC评测的基础上，进一步考虑了聚类后每个信息子集大小影响，调整权重，是最为适应本文任务的评测指标。

对信息全集T中的第i个信息T_i，若属于实验结果集的E^*和标准结果K^*，则对于该单个信息的准确率和召回率为$Precision_i=\dfrac{\#E^*\text{中与}\ T_i\ \text{重复的信息}}{|E^*|}$，$Recall_i=\dfrac{\#E^*\text{中与}\ T_i\ \text{重复的信息}}{|K^*|}$，再考虑各个信息的比重$w_i=1/|T|$有：

$$Precision_{B\text{-}cubed}=\sum_i w_i\cdot Precision_i$$

$$Recall_{B\text{-}cubed}=\sum_i w_i\cdot Recall_i$$

3.1 事件信息抽取系统

我们针对事件信息抽取系统来完成一个事件信

息的去重工作，对于事件信息，包含事件元素三元组{时间，地点，事件名}，因为该系统主要针对音乐会，演唱会，体育比赛等集会信息，因而时间，地点是该集会发生的时间和地点，事件名则是其名称。

系统共抽取到38 195个事件，从中选取发生时间在2009年9月的事件做实验数据集，剔除错误的抽取共得到1 022个事件。之所以选择相对接近时间的事件，是因为更有可能产生重复信息对。人工标注出其中的冗余关系，区分出其中的冗余事件对，包括有多条事件信息间的重复。NE信息（这里指事件信息的地名）的扩充，由于是无监督的方法实现，因而在38 195个事件全集上进行。

事件信息元素的相似度计算如下：

时间：时间属于格式化信息元素，尽管事件信息抽取和规范化后，时间都表示为[年：月：日：时：分：秒]的格式，但考虑到从不同网页上抽取来的事件在时间的具体节点上（比如时分）上必然有不一致，同时有些网页未必会标出具体时分，导致我们在存储时只能以0赋值，因此在计算时间元素的相似度时，我们只考虑年、月、日，若两个时间的年、月、日相同，则该维元素的相似度为1，否则为0。

地点：地点属于命名实体信息元素，我们会维护一个统一的地名表，只要是查表后得出两个地名元素指同一处，则该维元素的相似度为1，否则为0。但是这样的同义地名表毕竟有限，我们实验中使用前述的NE信息元素自动扩展的办法来扩充同义地名表。

事件名：事件名属于自由文本信息元素，我们采用向量空间模型，用TF-IDF作权重，来计算两个事件名之间的相似度，另外也采用字符串编辑距离（Levenshtein Distance，LD）。通过对事件名的观察，通常抽取得的事件名上都会有具体的小标题，如戏剧名，电影名，由《》或“”号括出，可以用模板匹配的方法找出事件标题中的小标题，小标题往往成为事件信息的重点，我们也用前述向量空间模型和字符编辑距离分别计算相似度作为特征。

使用以上方式计算事件信息各个元素的相似度作为特征，分别用线性分类器，和SVM分类器进行两两分类，并将有同类关系的事件进行聚合，作为最后的结果。表2是事件信息抽取系统中做冗余判断用到的特征。

表2　事件冗余判断的特征选择

时间是否为同一天，取值{0,1}	标题的相似度(TF-IDF)，取值[0,1]
地点在地名表中是否同义，取值{0,1}	标题的相似度(LD)，取值[0,1]
地点的相似度(TF-IDF)，取值[0,1]	标题中小标题的相似度(TF-IDF)，取值[0,1]
地点的相似度(LD)，取值[0,1]	标题中小标题的相似度(LD)，取值[0,1]

数据集：1 000个人工标注并聚类过的事件，我们将其分为两组，交叉验证，取平均结果。

表3　事件信息抽取系统去重实验

	Pairwise			MUC			B^3		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Linear-init	0.747 9	0.950 3	0.835 6	0.918 8	0.954 6	0.936 2	0.863 7	0.948 7	0.903 9
Linear-exp	0.793 7	0.917 9	0.838 6	0.932 6	0.965 5	0.948 5	0.878 4	0.958 4	0.914 9
SVM-init	0.740 4	0.998 5	0.834 9	0.945 5	0.995 3	0.969 6	0.878 0	0.994 9	0.930 7
SVM-exp	0.760 7	0.996 3	0.849 2	0.950 0	0.993 7	0.971 1	0.889 0	0.993 5	0.935 9

表3中，Linear指线性分类器的结果，后两行则是SVM的结果。后缀init给出的是初始的事件去重结果，分别用Pairwise，B^3和MUC指标来评测结果，exp则是指对事件地名做扩展之后的结果。

比较来看，Pairwise只表征两两信息对之间的结果，这一指标上差别不明显，SVM的结果要好于线性分类器的结果，特别是在指代消解的任务指标B^3和MUC上，F值分别对应地提高了3%～4%。同时，对应于线性分类器和SVM，exp的结果都好于init的结果。可以看出地名扩展在各个方法中，对信息的去重效果都有提升。

表4是一些扩充出来的地名对范例（每行的地名同义）：

表4 扩展地名对范例

逸夫舞台	天蟾逸夫舞台		
中山公园	中山音乐堂		
中国福利会少年宫	小伙伴剧场	中福会少年宫	上海小伙伴剧场
上海戏剧学院	上戏剧院	戏剧学院实验剧场	
可当代艺术中心	凯旋路613号B座		
新光小剧场	新光影艺苑		

从表格中可以看出，一些地名能够从文本相似的方法中判断出相似度，但是也存在文本不一致的情形，无法从文本的相似出发来解决是否冗余问题，这些也正是本文命名实体信息元素扩充的意义所在。地名的同义大致概括一下可以分为几类：多名，别称，整体部分场所，地址和地名等。

3.2 论文索引的去重

科研论文中包含大量的论文引用，抽取这些论文引用，可以分析出论文间的引用关系，研究方向的转移等，但论文引用的书写有许多重复，需要去重。

Cora数据集[8]，是已分析出重复论文引用的数据集，共包含1 878条引用记录。每个论文引用被抽取为一条信息，由于所细分的信息元素很多，且存在大量信息元素的混乱缺失，我们只抽取和使用以下信息元素：{author, title, venue, volume, address, pages, year}。该数据集上我们选择作者(author)一维NE元素进行扩充，同样在实验中分别给出扩充前后的结果。Cora数据集分为三组，我们分别使用一组训练，二组测试，这样取三次实验的平均结果如表5所示。

表5 Cora数据集的去重实验

	Pairwise			MUC			B^3		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Linear-init	0.858 4	0.996 6	0.920 3	0.975 0	0.996 5	0.985 6	0.898 2	0.9930	0.943 0
Linear-exp	0.858 5	0.996 6	0.920 4	0.975 5	0.996 5	0.985 8	0.898 9	0.993 0	0.943 4
SVM-init	0.874 6	0.976 2	0.922 1	0.982 7	0.997 1	0.989 9	0.926 3	0.990 5	0.957 3
SVM-exp	0.912 0	0.976 3	0.942 7	0.983 7	0.997 5	0.990 6	0.944 5	0.992 0	0.967 5

同样的，表5中分别用线性分类器linear和SVM来对Cora数据集进行实验。init是初始结果，而exp则是对Cora数据集中论文作者这一项进行自动扩展之后的结果，扩展之后结果也有一定的提升。在Cora数据集上，我们也采用线性分类器和SVM两种方法，SVM的结果要好于线性分类器，Linear的结果中，NE元素自动扩充的结果提升不明显，而SVM的方法使NE元素扩充对结果提升较大。分析可能原因是英文人名并不像事件信息中的中文地名那样难以处理，可能的变化就是缩写与全称，姓和名的交换等，而这些是能够通过文本相似度的方法来较好解决的。

文献[11-12]使用马尔可夫逻辑网(MLN)来对Cora数据集进行分段和指代消解(Segmentation and Entity Resolution)，由于Cora数据集分段(划分信息元素)时本身有错误，所以能够修正数据集本身的问题，最后文献[11]给出了Pairwise的准确度97.0%和召回率94.3%。

4 相关工作

信息的冗余判断和去重在数据库、数据整合、信息抽取等方面已有大量研究，这里列举一些和本文关系较大的工作。

信息去重在数据库中最早被定义为Record Linkage问题[13]，即将数据库中潜在的链接(相同记录)寻找出来，数据库的这种键值存储方式与本文的多元素信息十分相似，所以很多数据库方面的算法和比较函数都可以直接应用到冗余信息去重中。

文献[1]也是针对多元信息去重问题，定义了若

于相似度判断函数，应用到多元信息的每一维上，从而计算出两条信息记录之间在各维的相似度，并作为特征，利用 SVM 分类器得到信息记录间冗余与否的判断。文献[1]对不同类型的信息元素使用了同样的相似度计算函数，实际情况是，不同的信息元素在相似度判断上有简有难，比如时间表达式的相似判断比较简单，如果应用文献[1]中的计算方法，不仅复杂化，而且没能利用到时间表达式的格式化信息，效果反而不好；而命名实体信息仅靠文本相似度来判断又不能很好解决。本文对信息元素大致分为四类，由相似度判断的难易不同，可应用不同的计算函数；特别对命名实体信息元素，在传统文本相似处理不适用时，应用同义实体自动扩展的方法，从语料中学习出一些同义实体，从而提高整个信息去重的效果。

文献[14]提出了域匹配(Field Matching)的想法，来进行信息的去重，文中的域指字符串，多针对DNA、蛋白质检测等生物信息的应用，而每个域又由子域构成，可以从子域的相似判断到信息的相似判断。将域切分成子域，进而去重的想法与本文先做信息抽取后去重的方法比较类似，只是网页中信息的各个抽取点不像[14]的应用环境那么连续，中间有广告，标签，图片等无关内容。

文献[15]通过定义特定信息元素之间的转移来计算两条信息之间的相似性，尽管最后实验结果不错，但也定义和使用了大量复杂的转移规则，并且规则仅能应用于特定域，在实际使用中需要大量领域知识等，不容易推广。

文献[16]使用 LDA 模型，特别针对命名实体类的信息元素做信息去重，而且对去重后的每个信息类都能够给出一个代表信息，即对类内的所有信息提出一个统一的代表信息，这对信息抽取系统最后的展示十分重要。

文献[2-3,17]都是以提升信息去重的效率为目的，当问题规模增大以后，需要考虑到效率问题，特别是对类似本文的两两比对办法。采用分块的办法，快速提取可能有重复的信息对，可以避免不必要的比对，文献[2-3]都是传统的分块方法，需要信息抽取中的某些信息元素对重复与否有很明显的指导作用。文献[17]针对某一些信息元素可排序的条件进行分块，在一些信息抽取任务中会有比较好的效果。

5　总结

在冗余信息去重问题中，本文针对各个信息元素上相似度判断的不一致性，将其分为四类分别处理，这样既可以利用各信息元素特有的格式信息，又能使各种相似度函数在不同信息元素上的应用变得独立，改变以往对所有信息元素统一处理造成的混乱。同时，对比较依赖词典等外部知识的命名实体信息元素，利用信息间其他元的相似性，进行同义实体的自动扩充，无论是扩充结果还是最后去重结果，都可以证明这种自动扩充是有效的。

参考文献

[1] Mikhail Bilenko, Raymond J. Mooney. Adaptive Duplicate Detection Using Learnable String Similarity Measures [C]//Proceedings of KDD, Washington, DC, USA, 2003: 39-48.

[2] Rohan Baxter, Peter Christen, Tim Churches. A Comparison of Fast Blocking Methods for Record [C]//Proceedings of KDD. Washington, DC, USA, 2003: 25-27.

[3] Lifang Gu, Rohan Baxter. Adaptive Filtering for Efficient Record Linkage[C]//Proceedings of the Fourth SIAM International Conference on Data Mining, Lake Buena Vista, Florida, USA, 2004: 477-481.

[4] 李峰，李芳. 中文词语语义相似度计算——基于《知网》2000[J]. 中文信息学报，2007，21(3)：99-105.

[5] 王荣波，池哲儒. 基于词类串的汉语句子结构相似度计算方法[J]. 中文信息学报，2005. 19(1)：21-29.

[6] 张奇，黄萱菁，吴立德. 一种新的句子相似度度量及其在文本自动摘要中的应用[J]. 中文信息学报，2005，19(2)：93-99.

[7] William W. Cohen, Pradeep Ravikumar, Stephen E. Fienberg. A comparison of string distance metrics for name-matching tasks [C]//Proceedings of IJCAI, 2003: 73-78.

[8] http://www. cs. umass. edu/～ mccallum/code-data. html [OL].

[9] M Vilain, J Burger, J Aberdeen, et al. A model-theoretic coreference scoring scheme[C]//Proceedings of the 6th Conference on Message Understanding. Columbia, Maryland, USA, 1995: 45-52.

[10] Amit Bagga, Breck Baldwin. Algorithms for Scoring Coreference Chains[C]//Proceedings of The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference. 1998: 563-566.

[11] Hoifung Poon, Pedro Domingos. Joint Inference in Information Extraction[C]//Proceedings of the 22nd National AAAI Conference on Artificial Intelligence. Vancouver, British Columbia, Canada, 2007: 913 - 918.

[12] Hoifung Poon, Pedro Domingos. Joint unsupervised coreference resolution with Markov logic[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Honolulu, Hawaii, USA, 2008: 650-659.

[13] Ivan P. Fellegi, Alan B. Sunter. A Theory for Record Linkage[J]. Journal of the American Statistical Association, 1969, 64(328): 1183-1210.

[14] Alvaro Monge, Charles Elkan. The field matching problem: Algorithms and applications[C]//Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. 1996: 267-270.

[15] Steven N. Minton, Claude Nanjo, Craig A, et al. A heterogeneous field matching method for record linkage[C]//Proceedings of the 5th IEEE International Conference on Data Mining. Houston, Texas, USA, 2005: 314-321.

[16] Indrajit Bhattacharya, Lise Getoor. A Latent Dirichlet Model for Unsupervised Entity Resolution [C]//Proceedings of the Sixth SIAM International Conference on Data Mining. Bethesda, MD, USA. 2006: 47-58.

[17] Akiko Aizawa. A fast linkage detection scheme for multi-source information integration[C]//WIRI, Tokyo, Japan, 2005: 30-39.

(上接第41页)

[9] Kishore Papineni, Salim Roukos, Todd Ward, et al. Bleu: a method for automatic evaluation of machine translation [C]//Proceedings of ACL 2002. 2002.

[10] Deyi Xiong, Qun Liu and Shouxun Lin. Maximum Entropy Based on Phrase Reordering Model for Statistical Machine Translation [C]//Proceedings of ACL 2006,2006.

[11] Zhongjun He, Qun Liu, Shouxun Lin. Improving statistical machine translation using lexicalized rule selection[C]//Proceedings of EMNLP 2008, 2008.

[12] Zhongjun He, Yao Meng, Hao Yu. Maximum Entropy Based Phrase Reordering for Hierarchical Phrase-based Translation [C]//Proceedings of EMNLP 2010, 2010.

[13] Franz Josef Och, Hermann Ney. A systematic comparison of various statistical alignment models[J]. Computational Linguistics, 2004, 29(1): 19-51.

[14] Andreas Stolcke. 2002. Srilm-an extensible language modeling toolkit [C]//Proceedings of the 7th International Conference on Spoken Language Processing. 2002: 901-904.

[15] Franz Joseph Och. Minimum error rate training in statistical machine translation [C]//Proceedings of ACL 2003. 2003.

[16] Yang Liu, Qun Liu, Shouxun Lin. Tree-to-String Alignment Template for Statistical Machine Translation [C]//Proceedings of ACL 2006. 2006.

[17] Michel Galley, Jonathan Graehl, Kevin Knight, et al. Scalable Inference and Training of Context-Rich Syntactic Translation Models [C]//Proceedings of ACL 2006. 2006.