doi:103969/j. issn. 0490-6756. 2011. 02. 012

中文财经文本中公司名简称的自动识别

陈　超[1]，朱洪波[1]，王亚强[1]，韩国辉[1]，谭　斌[2]，于中华[1]
(1. 四川大学计算机学院，成都 610065；2. 四川大学锦江学院，彭山 620860)

摘　要：命名实体识别是当前自然语言处理的热点问题之一，对信息检索、信息抽取等具有重要意义. 然而，目前多数研究都集中在对命名实体全称的识别上. 本文以财经为领域背景，对从文本中识别简称，并将其映射成全称问题进行了研究，提出了一个启发式算法用于解决该问题. 所提出的算法首先提取文本中每个 N 元组(N-gram)作为候选的公司名简称，然后建立 n 元组与全称表中每个全称的最优对齐关系，最后对每对“N 元组-全称”对齐关系进行评价和筛选，识别出文本中的简称及每个简称对应的全称. 在随机获取的网页文本集上对所提出的算法进行了实验测试，算法的精确率、召回率和 F-度量值分别为 83.62%、87.28%、85.41%.

关键词：命名实体识别；公司名；简称；启发式
中图分类号：TP391　　**文献标识码**：A　　**文章编号**：0490-6756(2011)02-0308-07

Automatic recognition of company name abbreviations in Chinese financial texts

CHEN Chao[1], *ZHU Hong-Bo*[1], *WANG Ya-Qiang*[1], *HAN Guo-Hui*[1], *TAN Bin*[2], *YU Zhong-Hua*[1]
(1. College of Computer Science, Sichuan University, Chengdu 610065, China;
2. SCU Jinjiang College, Pengshan 620860, China)

Abstract: At present, Named Entity (NE) recognition is one of the hot problems in Natural Language Processing (NLP), and plays a significant role in information retrieval and information extraction. However, the majority of studies were concentrated on NE full name recognition. Taking financial as the example domain, this paper studied the problem of recognizing financial NE abbreviations in texts and mapping the abbreviations to their corresponding full names, and proposed a heuristic algorithm to solve the problem. The algorithm at first extracted every n-gram from a text as a candidate of a company name abbreviation, then established the optimal alignment between the candidate and every company full name in a full name list, and finally recognized the candidate as an abbreviation and mapped it to its full name based on evaluating and filtering heuristically the alignments. The experiments preformed on a text set obtained randomly from the Web showed that the precision, recall and F-score of the algorithm reach 83.62%、87.28% and 85.41% respectively.

Key words: named entity recognition, company name, abbreviation, heuristic

收稿日期：2010-07-07
作者简介：陈超(1985—)，男，硕士研究生，研究方向为数据挖掘与自然语言处理.
通讯作者：于中华. E-mail：yuzhonghua@scu. edu. cn

1 引　言

在经济全球化的大背景下，以文本作为载体的财经类信息的数量越来越大，如何有效地对其进行归纳、分析和整理成为当前学术界和产业界都关注的一个热点问题. 在财经类文本中，公司名往往承载着所表达的关键信息，对其识别是财经文本挖掘的关键. 然而，由于公司命名的用字规律不强，使用比较随意，经常以简称的形式出现，如“中国石油化工集团公司”经常简称为“中国石化”或“中石化”，这都为公司名的识别带来了困难. 据我们统计发现，在从网易财经随机获取的 100 篇金融新闻文本中共出现了 1996 个公司名，其中全称仅 148 个，简称却有 1848 个，前后文中有全称伴随的简称仅有 374 个. 因此对简称高效、准确的识别对充分挖掘利用财经文本信息，具有重要意义.

2 相关工作

近年来，研究者对公司名及其他命名实体的识别已经提出了一系列的算法，并且其中的一些算法已经可以达到很高的识别准确率，如基于隐马尔可夫模型的生物医学命名实体识别和分类[1]，基于角色的中文命名实体识别[2]等，然而，对公司名简称识别的研究成果，目前还比较少见. 据我们所知，仅有的几项相关工作包括：文献[3]以人工总结的、刻画公司名构成及其上下文信息的六个知识库为基础，通过两次扫描匹配实现对文本中公司名的识别. 在开放测试中达到的精确率和召回率分别为 62.8%和 62.1%. 该文重点在于识别全称，因此对简称的识别非常简单，只粗略总结了四种全称缩略的情况，据此制定匹配规则，而且算法的精确率和召回率都偏低. 文献[4]利用简称在文本中第一次出现时伴随的全称信息，提出了基于规则的算法用于识别简称，算法取得了比较好的效果，然而财经新闻、网上评论等非官方形式的文本中，简称的首次出现很少伴随有全称，这使得算法的使用范围大大受限. 文献[5]提出了一种基于隐马尔可夫模型(Hidden Markov Model)的算法用于对中文文本中缩略语的识别和理解. 文中对缩略语映射到全称和全称生成缩略语分别进行了测试，精确率分别为 51%和 72%. 该算法虽然模型优美，使用的启发式信息少，然而算法假定全称中每个词都至少有一个汉字出现在简称中，这限制了算法模型的处理能力[8]. 文献[6]通过规则和贝叶斯概率模型来识别全称，在全称的基础上利用一些规则来识别简称. 算法对简称识别的平均精确率和平均召回率分别为 74.14%、67.18%，而对金融机构名简称的识别却只能达到 70%的精确率和 58.33%的召回率. 文献[7]提出了一种基于 SVR(支持向量回归)的打分方法来预测短语缩略语. 结果表明，所提出的算法优于 HMM 和启发式方法. 然而，该算法的任务是根据全称预测简称，不是识别简称并确定其对应的全称，与本文的任务不同. 文献[8]提出了基于条件随机场的机构名简称生成模型，取得了优于文献[7]的效果. 然而，正如文献[9]所说，由于公司名简称的长度和产生规则并没有统一的规范，因此该算法的效果容易受到训练语料的影响. 文献[9]也采用条件随机场对全称生成简称的规律进行了研究.

上述现有的工作，有些没有考虑公司名简称的识别，有些虽然考虑了，但是处理过程比较简单，准确率也偏低，或者仅仅处理了简称与全称共现的情况，而另外一些工作仅仅研究了全称生成简称问题，没有考虑如何从自由文本中抽取简称并映射到全称. 为此，本文对从文本中识别简称，并将其映射成全称问题进行了研究，提出了一个启发式算法用于解决该问题. 所提出的算法首先提取文本中每个 N 元组(N-gram)作为候选的公司名简称，然后建立 N 元组与每个全称的最优对齐关系，最后对每对“N 元组-全称”对齐关系进行评价和筛选，识别出文本中的简称及每个简称对应的全称. 对于任意一个 N 元组，如果所有的全称都被算法过滤掉，则判定给定的 N 元组不是一个公司名简称，否则说明它是一个简称，对应的全称是通过了筛选的所有全称中排在全称词表最前面的那个. 在随机获取的网页文本集上对所提出的算法进行了实验测试，结果表明，算法达到的精确率为 83.62%，召回率为 87.28%，F-度量值为 85.41%，同时明显优于文献[6]所提出的算法.

3 公司名简称的自动识别

3.1 问题描述

本文要解决的问题可以形式化地描述为：假定 $D = \{NE_1, NE_2, \cdots, NE_n\}$ 为公司名全称词典，对于文本中的任意字串 $l_1 l_2 \cdots\cdots l_m$，建立映射关系

$$f: l_1 l_2 \cdots l_m \rightarrow x \in D \cup \{\varphi\}$$

字串 $l_1 l_2 \cdots\cdots l_m$ 被映射成 φ 表示它不是公司名的简称，否则它是简称，$x \in D$ 为其全称.

本文后续分析和算法设计基于如下合理的假

设:(1) 简称中的每个汉字均来自对应的全称;(2) 简称中汉字的先后顺序与它们在全称中的顺序相同.虽然存在违反上述假设的公司名和金融机构名,如"中国人民银行"可以缩写为"央行","中国第一核能发电厂"的简称为"核一厂",然而据我们观察,这种情况非常少见.

3.2 算法基本思想

算法利用结构化的全称词典,基于全称与候选简称之间的对齐关系实现对候选的打分,并结合一些启发式规则实现简称的识别和简称到全称的映射.结构化全称词典中的每个全称被划分为地名、公司名核心串和公司名后缀三部分.由于手工对原始的全称表进行结构化工作量较大,本文设计了算法实现对全称的自动划分.

3.3 自动结构化全称词典

一般来说,公司名的结构具有一定的规律性,一个完整的公司名可以划分为地名、公司名关键字、公司类型和公司名后缀4部分[3],其中公司名关键字和公司类型可能省略.据观察,在形成简称时,全称不同部分的用字进入简称的机会各不相同,因此需要对各部分的用字区别对待.本文设计了算法实现对全称词典中全称的自动划分,首先划分成地名、公司名核心串和公司名后缀三部分,如"中国石油化工集团有限公司"划分为"中国/石油化工集团有限/公司".划分时算法使用了地名库和公司名后缀库,前者包括中国各个省、市的名称,后者包括"公司"、"银行"、"集团"、"局"、"所"、"厂"等后缀.将公司名首先划分成三部分而非四部分的原因,主要是本文后续算法的需要.

根据观察,一个具有核心串的全称在形成简称时,该核心串中某些词的用字一定会在简称中出现.进一步地,在形成简称时,核心串中的关键字和公司类型进入简称的可能性各不相同,一般来说,简称中出现全称关键字的可能性远远大于出现公司类型的可能性.因此,如果能够对核心串进行分词,判断核心串是否具有关键字,并在有关键字的情况下切分出关键字,对于识别简称将带来很大的便利.然而,核心串中的关键字往往是专有名词,很少出现在词典中,传统的基于词典的方法难以用于对公司名核心串的分词,并识别出其中的关键字.因此,本文提出了一个基于Bi-gram频率的算法,用于从核心串中切分关键字和公司类型,并对核心串进行分词.算法首先将核心串切分成二元字串(Bi-gram)序列,然后针对每个Bi-gram,统计全称词典中包含该Bi-gram的不同全称的个数,这样每个全称被切分成形如$<B_1, C(B_1)>, <B_2, C(B_2)>, \cdots, <B_m, C(B_m)>$的序列,其中$B_i$为一个Bi-gram,$C(B_i)$为它的频率(即包含它的不同全称的个数),$i = 1, 2, \cdots, m$.最后扫描该序列,寻找满足如下条件的一组汉字序号$l_1 = 1 < l_2 < \cdots < l_r$ = 核心串长度+1,作为对核心串的切分位置(即切分出的每个词首汉字的序号).

(1) $\forall k = 1, 2, \cdots, r-1$,如果$l_{k+1} - l_k > 1$,则核心串第$l_k$到第$l_{k+1} - 1$个汉字之间的所有Bi-gram具有相同的频率;

(2) $\forall k = 2, \cdots, r-1$,由第$l_{k-1}$和$l_k$个汉字构成的Bi-gram的频率一定小于由第$l_k$和$l_{k+1}$个汉字构成的Bi-gram的频率.

本文分别设计了正向和逆向两个算法,来搜索满足上述条件的切分位置.算法1描述了正向切分的过程,逆向版本与此类似,只是处理方向是从核心串的尾部开始.

算法1: 核心串正向切分算法

输入: $<B_1, C(B_1)>, <B_2, C(B_2)>, \cdots\cdots, <B_m, C(B_m)>$

输出: 切分位置$l_1 = 1 < l_2 < \cdots < l_r$ = 核心串长度+1

```
Begin
    Finish = False; i := 2; Curr := B1;  C := C(Curr); //C 为 Curr 中二元组的频率
    while (not Finish) do
        while (i <= n and C == C(Bi))  do
        //如果下一个 Bi-gram 和 Curr 中 Bi-gram 的频率相同
        Curr:=Curr + Bi; i++;//下一个 Bi-gram 与 Curr 中的 Bi-gram 合并成词
        end while
        if (i <= n) //下一个 Bi-gram 的频率与 Curr 中 Bi-gram 的频率不同
        设 Curr 中最后一个二元字串 Bi-1 =L0L1;Bi = L1L2;//其中 L0,L1 和 L2 是汉字
```

```
if (C(Curr) < C(B_i))
    Curr := Curr - L_1; 输出 Curr;//为切分出的一个词
    Curr := B_i; C_i := C(Curr); i ++;//Curr 中二元组的频率
else
    输出 Curr; i ++;//为切分出的一个词
    if (i <= n)
        Curr := B_i; C_i := C(Curr); i ++;//Curr 中二元组的频率
        else
            切分 L_2 为词; Finish = true;//扫描完所有二元组
        end if
    end if
    else
    Finish := true;
    end if
end while
End
```

一般来说，一个全称如果具有关键字，则一定会出现在核心串的头部. 因此，可以采取简单的如下方法来判断一个全称是否具有关键字，关键字是什么. 即如果从全称核心串切分出的第一个词在全称词典中出现的次数低于预先指定的阈值，则认为该词为相应全称的关键字，否则认为该全称没有关键字.

经过上述处理，全称词典中每个全称被结构化地表示成四元组(地名，核心串词序列，后缀，是否有关键字)，其中核心串词序列由上述算法生成，“是否有关键字”为布尔型量，表示核心串中是否包含公司名关键字(如果有，则核心串的首词被认为是其关键字). 例如，全称“云南云天化集团有限公司”被结构化地表示为(云南，云天化/集团/有限，公司，True).

此外，有些全称中的词，由于它区分不同公司名的能力很弱，出现在简称中的可能性几乎没有. 如果能搜集到这些词，构成禁止词表，则在简称向全称映射时，可以忽略这些词，这样不但可以提高简称识别和向全称映射的效率，而且可以降低误识别的风险. 然而，手工构造和维护这样的禁止词表异常繁琐单调，而且工作量大，为此，本文借鉴 IDF (Inverse Document Frequency)的思想，用 IFNF (Inverse Full Name Frequency)来评估全称中字串区分不同全称的能力. IFNF 定义为

$$IFNF_w = N_f / N_w$$

其中 w 为全称中出现的 Bi-gram，N_f 为全称词典中全称的总数，N_w 为全称词典中包含 w 的全称总数. 若 w 满足条件

$$IFNF_w = N_f / N_w < \theta,$$

θ 为设定的阈值，则把 w 加入禁止词表. 在识别简称时，首先判断从文本中提取的候选串是否在禁止词表中，如果是，则直接判断该串不是简称，否则再做进一步的处理.

3.4　n 元字串(候选简称)与全称的最佳对齐

设 $A = a_1 a_2 \cdots a_m$ 为文本中的一个字串，$S = s_1 s_2 \cdots s_n$ 为全称表中的一个全称，为了判断 A 是否为 S 的简称，首先建立 S 和 A 之间的最优对齐关系，并把这种对齐关系用序列化 0-1 标注串来表示. 设 $P = p_1 p_2 \cdots p_n$，$p_i \in \{0,1\}$，$i \in [1,n]$，$p_i = 1$ 表示 s_i 在缩写过程中被保留，$p_i = 0$ 表示 s_i 被删除. 这里通过编辑距离(Edit Distance)算法[10]得到的 S 和 A 的距离矩阵 ED 来确定它们的最优对齐关系，进而得到序列化标注串 P.

根据前文假设，全称缩写为简称时不可能有字符替换和插入操作，因此在求序列化标注串 P 前，可以先判断 S 转化为 A 的过程是否存在替换或插入操作. 如果存在，则直接得出结论：A 不是 S 的简称；否则，采用如下所示的迭代过程计算序列化标注串 P.

初始化：

$$i = n, \text{next } j == m, j = m$$

$$p_i = \begin{cases} 1, & \text{如果 } s_i = a_j \\ 0, & \text{否则} \end{cases}$$

递推:对于 $i=n-1,\cdots,1$

$$j=\begin{cases}\text{next } j-1,\text{如果 } s_{i+1}=a_{\text{next } j}\\ \text{next } j,\text{如果 } s_{i+1}\neq a_{\text{next } j},\text{且 } ED_{i,\text{next } j}<ED_{i,\text{next } j-1}\\ \text{next } j-1,\text{其他}\end{cases}$$

$$pi=\begin{cases}1, & \text{如果 } s_i=a_j\\ 0, & \text{否则}\end{cases}$$

$$\text{next } j=j$$

迭代结束,就得到序列化标注串 $P=p_1p_2\cdots p_n, i\in[1,n]$.

3.5 权重计算与筛选

经过观察分析,在形成公司名简称的过程中,地名和公司名后缀的重要性相当,它们的用字进入简称的机会接近,而核心串用字进入简称的可能性远远高于前两者.因此,可以对全称的三段(地名、核心串、后缀)分别赋以不同的权重,以区分它们在形成简称过程中的重要性.此外,对于核心串部分,越排在前面的汉字进入简称的可能性一般越大,因此本文对核心串每个汉字赋予不同的权重,排在前面的汉字的权重与排在后面的汉字的权重相比有一个特定的增量.在上述加权模式下,对每一对 S 和 A 的序列化标注串(表示它们之间的最优对齐关系),本文采取如下方法计算相应对齐关系的得分.

设公司名全称 $S=s_1\cdots s_{d_1}/s_{d_1+1}\cdots s_{d_2}/s_{d_2+1}\cdots s_n$,其地名、核心串和后缀各段的权值分别为 v_1,v_2 和 v_3(人工预先设定,且满足 $v_2>v_1=v_3$;如果一个全称缺失某段,则对于该全称其对应段的权值设定为0),$\text{AllScore}(S)=v_1+v_2+v_3$ 为 S 的总权值,A 与 S 最优对齐后各段的得分分别为 Seg_1,Seg_2 和 Seg_3,A 与 S 匹配的总得分为 $\text{Score}(A,S)$.另设全称核心串(即 $s_{d_1+1}\cdots s_{d2}$)最后一个汉字的权值为1,核心串每个汉字相对于后一个汉字的权值增量为 c,则核心串倒数第 i 个汉字的权重为 $(1+c)^{i-1}$,核心串的权值总和为 $L=\sum_{i=1}^{d_2-d_1}(1+c)^{i-1}$,$A$ 与核心串的匹配得分为 $l=\sum_{i=1}^{d_2-d_1}p_{d_2-i+1}(1+c)^{i-1}$,其中 P_i 是 S 和 A 最优对齐的序列化标注串相应位的值,为0或1.简称候选 A 与全称 S 匹配的总得分定义为

$$\text{Score}(A,S)=Seg_1+Seg_2+Seg_3$$

其中地名得分为

$$Seg_1=\begin{cases}v_1, & \text{如果}\sum_{i=1}^{d_1}p_i>0\\ 0, & \text{否则}\end{cases}$$

核心串得分为

$$Seg_2=\begin{cases}v_2, & \text{如果 } l/L>\varepsilon,0<\varepsilon<1\\ 0, & \text{否则}\end{cases}$$

后缀得分为

$$Seg_3=\begin{cases}v_3, & \text{如果}\sum_{i=d_2+1}^{n}p_i>0\\ 0, & \text{否则}\end{cases}$$

设定过滤阈值为 φ,如果候选 A 匹配 S 的得分满足条件 $\text{Score}(A,S)/\text{AllScore}(S)>\varphi$,则将进行进一步的过滤,反之,不能成为简称.

3.6 过滤策略

上述过滤策略在识别简称和向全称映射时可能存在如下两个困难:(1) 可能将文本中相同位置开始的若干个串识别成相同全称的简称,如从文本"中石化集团的海外市场得以开拓……"中可能同时识别出"中石化集团"和"中石化",在这种情况下必须选择其一;(2) 上述算法没有利用中文语义表达上的规律性,只是单纯地建立在字符匹配的基础上.为了进一步提高算法识别的精度,提出了如下的过滤策略.

3.6.1 最长序列优先 对于3.6节中第(1)种困难,总是优先选择最长的简称,即对于上例的文本,认为其中存在简称"中石化集团",而不是"中石化".

3.6.2 缩写表义规则 通过观察和分析得出如下3条中文缩写表义规则:(1) 如果全称中没有关键字,则简称中一定要出现地名或地名中的汉字,例如"中国投资有限责任公司",可以缩写为"中投公司"或"中投",但如果没有地名中的汉字"中",相应的缩写就不符合中文的表达习惯了;(2) 如果全称中有关键字,那么其简称中必定会出现关键字的全部或其部分汉字;(3) 简称中任意相邻两个单字必须属于核心串的同一个单词或相邻两个单词,并且连续单字的个数必须是偶数.如果核心串只缩写成一个单字,则地名必须也缩写成一个汉字.

缩写表义规则中的"单字"是同时满足如下4个条件的全称第 i 个汉字:(1) $p_i=1$;(2) $i=0$ 或者 $p_{i-1}=0$;(3) $p_i=n$ 或者 $p_{i+1}=0$;(4) 该汉字属于核心串.

例如,"中国/南方/航空/集团/公司"的一个缩写为"中国南航",其中的单字"南""航"配对且它们各自的原词"南方"和"航空"相邻;"中国/国际/金融/有限/公司"的一个缩写为"中金公司",其中单

字“中”和“金”分别来自地名和核心串.

对于给定的 A，如果一个 S 不满足上述缩写表义规则，则判定 S 不是 A 的全称.

4　实验结果与分析

本文的实验数据是来自网易财经的金融文本，共 167 篇，分为两组数据，第一组 67 篇文本用于算法设计；第二组 100 篇文本用于算法测试. 公司名全称表是人工收集的，保证文本中简称所对应的全称，表中一定存在. 表 1 给出了在如下参数设置下的结果：$v_1 = 1, v_2 = 3, v_3 = 1, c = 1.0, \varepsilon = 0.25, \varphi = 0.5, \theta = 150.0$，核心串分词扫描方式为逆向.

为了衡量各个方法和策略对实验结果的影响，在相同参数下，针对第二组数据集做了第二个实验来看简称禁止词表和缩写表义规则对结果的影响.

表 1　不同数据集的实验结果

Tab. 1　The performance on different datasets

数据分组	缩写总数	精确率(%)	召回率(%)	F 值(%)
第 1 组	1298	88.36	92.99	90.62
第 2 组	1848	83.62	87.28	85.41

记权重计算筛选为基础方法 BASE，简称禁止词表策略为 NFP，缩写表义规则过滤为 ASFP. 实验结果见表 2.

在权值计算筛选过程中，算法的重心在于公司名核心串的得分计算，而计算中假设了核心串中的字从后往前的权值是增量关系. 为了证明假设的合理性，针对第二组数据做了第三个实验，设定 $c = 0$ 和 $c = 1.0$，使用 BASE 方法所得到的结果如表 3 所示.

如上 3.3 节所述，核心串分词的扫描方式有两种：正向和逆向. 表 4 给出了正向和逆向对实验结果的影响. 为了进一步验证算法的效果，将本文算法与文献[6]所提出的算法在相同数据集上进行了测试，结果见图 1.

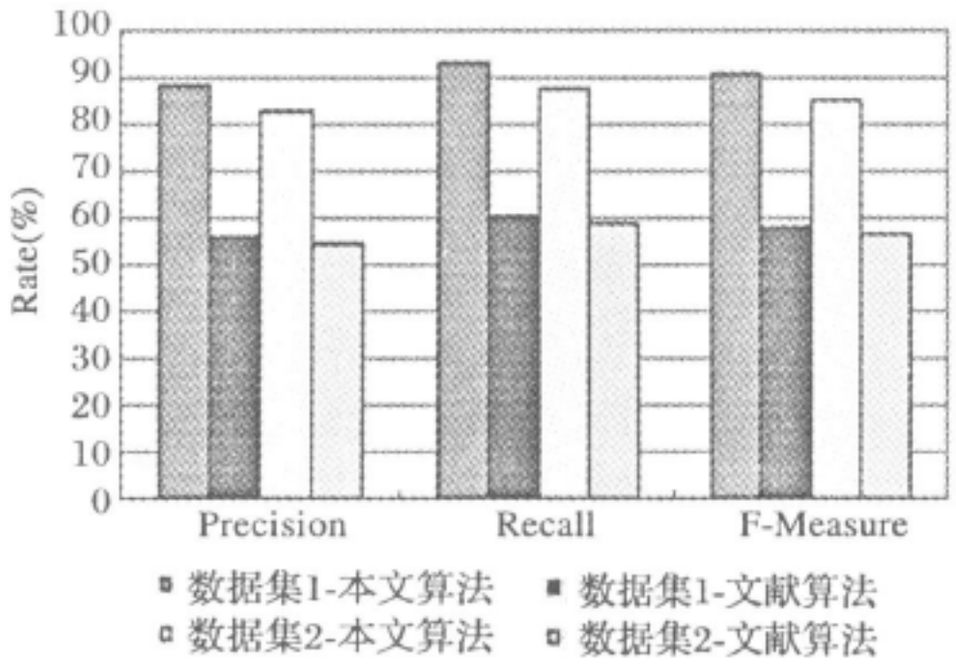

图 1　与文献[6]的实验结果对比

Fig. 1　Our algorithm vs. the algorithm in paper[6]

从表 1 可以看出本文设计的算法取得了不错的效果，实验结果也比较稳定. 表 2 说明了简称禁止词表对平衡精确率和召回率有较好的效果，缩写表义规则对提高精确率有非常好的效果，也说明了

表 2　各个策略对实验结果的影响

Tab. 2　The effect of strategies on final performance

使用方法	精确率(%)	召回率(%)	F 值(%)
BASE	44.12	98.48	60.94
BASE+NFP	59.40	93.72	72.71
BASE+ASFP	80.36	91.02	85.36
BASE+NFP+ASFP	83.62	87.28	85.41

表 3　权重 c 对实验结果的影响

Tab. 3　The effect of different values of the weight c on final performance

c 值	精确率(%)	召回率(%)	F 值(%)
$c = 0.0$	27.78	99.51	43.43
$c = 1.0$	44.12	98.48	60.94

表 4　核心串扫描方式对实验结果的影响

Tab. 4　The effect of different core-string scanning methods on final performance

扫描方式	精确率(%)	召回率(%)	F 值(%)
正向	86.16	85.88	86.02
逆向	83.62	87.28	85.41

中文缩写表达上确实有规律或规则可循.表5反映出本文对核心串从后往前的权重增量关系的策略是合理的.在表3中,核心串的正向和逆向扫描的分词结果对最终简称的判别并没有太大影响.图1反映出本文所提出的算法在精确率、召回率和F-度量值方面都明显好于文献[6]所提出的算法,其原因正如文献[6]所述,所构建的简称规则集很不完善,产生了过多冗余,也产生了很多遗漏.

算法错误识别的绝大多数情况是由公司名关键字识别错误引起的,如"中国对外投资发展有限公司"的"对外投资"被判断为关键字,因此它在文本中的出现都被标记为简称.漏识的情况,主要是由禁止词表造成的,如果某个公司有很多下属公司和合资公司,而所有这些公司全称中都包含该公司的关键字,则造成该关键字频率增大而被认为是常用词过滤掉.如"中国海尔集团"有"青岛海尔有限公司","海尔电器有限公司"等子公司,因此本可以作为简称的"海尔"被过滤掉.此外也有不满足核心串权重增量关系的情况,如"中国船舶重工集团有限公司"的简称"中国重工"就不满足这个关系.

5 结 语

公司是金融新闻文本中谈论的对象和主体,有效地识别其简称能为文本的信息抽取提供更好的基础.本文采用启发式方法识别金融文本中所存在的公司名简称,取得了不错的效果.下一步工作是对本文所提出的算法进行更广泛的测试,这既包括搜集更多的测试语料,也包括扩充全称词表,扩大算法识别公司名的范围.此外,利用文献中提出的简称生成模型来从文本中识别简称,也是进一步的工作方向.

参考文献:

[1] Shen D, Zhang J, Zhou G D, *et al*. Effective adaptation of a hidden markov model-based named entity recognizer for biomedical domain[C] // Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine. Sapporo, Japan: [s. n.], 2003.

[2] 俞鸿魁,张华平,刘群. 基于角色标注的中文机构名识别[J]. 计算机学报,2004,27(1):85.

[3] 王宁,葛瑞芳,苑春法,等. 中文金融新闻中公司名的识别[J]. 中文信息学报,2002,16(2):1.

[4] 张占英,王中立. 中文文本中公司名简称的识别[J]. 许昌学院学报,2003,22(2):99.

[5] Chang J S, Lai Y T. A preliminary study on probabilistic models for chinese abbreviations[C]. Barcelona, Spain: Association for Computational Linguistics, 2004.

[6] 沈嘉懿,李芳,徐飞玉,等. 中文组织机构名称与简称的识别[J]. 中文信息学报,2007,21(6):17.

[7] Sun X, Wang H F, Wang B. Predicting chinese abbreviations from definitions: an empirical learning approach using support vector regression[J]. Journal of Computer Science and Technology, 2008, 23(4): 602.

[8] Yang D, Pan Y C, Furui S. Automatic chinese abbreviation generation using conditional random field [C]. Boulder, Colorado: Association for Computational Linguistics, 2009.

[9] 陈茂松,陈秀群. 中国计算机语言学研究前沿进展(2007~2009)[C]. 北京:清华大学出版社,2009:546.

[10] Navarro G, Raffinot M. Flexible pattern matching in strings [C]. Cambridge, England: Cambridge University Press, 2001.

[责任编辑:伍少梅]