ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.19, No.3, March 2008, pp.557−564
DOI: 10.3724/SP.J.1001.2008.00557

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种基于判别式重排序的拼写校正方法*

张 扬[1+], 何丕廉[1], 向 伟[2], 李 沐[3]

[1](天津大学 计算机科学与技术学院,天津 300072)

[2](香港科技大学 计算机系,香港)

[3](微软亚洲研究院,北京 100080)

A Discriminative Reranking Approach to Spelling Correction

ZHANG Yang[1+], HE Pi-Lian[1], XIANG Wei[2], LI Mu[3]

[1](School of Computer Science and Technology, Tianjin University, Tianjin 300072, China)

[2](Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong, China)

[3](Microsoft Research Asia, Beijing 100080, China)

+ Corresponding author: Phn: +86-22-27402114, E-mail: yangzhang@tju.edu.cn, http://www.tju.edu.cn

Zhang Y, He PL, Xiang W, Li M. A discriminative reranking approach to spelling correction. *Journal of Software*, 2008,19(3):557−564. http://www.jos.org.cn/1000-9825/19/557.htm

Abstract: This paper proposes an approach to spelling correction. It reranks the output of an existing spelling corrector, Aspell. A discriminative model (Ranking SVM) is employed to improve upon the initial ranking, using additional features as evidence. These features are derived from state-of-the-art techniques in spelling correction, including edit distance, letter-based *n*-gram, phonetic similarity and noisy channel model. This paper also presents a method to automatically extract training samples from the query log chain. The system outperforms the baseline Aspell greatly, as well as the previous models and several off-the-shelf systems (e.g. spelling corrector in Microsoft Word 2003). The experimental results based on query chain pairs are comparable to that based on manually-annotated pairs, with 32.2%/32.6% reduction in error rate, respectively.

Key words: spelling correction; discriminative model; reranking; log mining; query chain

摘 要: 提出一种基于判别模型的拼写校正方法.它针对已有拼写校正系统Aspell的输出进行重排序,使用判别模型Ranking SVM来改进其性能.将现今较为成熟的拼写校正技术(包括编辑距离、基于字母的*n*元语法、发音相似度和噪音信道模型)以特征的形式整合到该模型中来,显著地提高了基准系统Aspell的初始排序质量,同时性能也超过了一些商用系统(如 Microsoft Word 2003)的拼写校正模块.此外,还提出了一种在搜索引擎查询日志链中自动抽取拼写校正训练对的方法.基于这种方法训练的模型获得了基于人工标注数据所得结果相近的性能,它们分别将基准系统的错误率降低了32.2%和32.6%.

关键词: 拼写校正;判别模型;重排序;日志挖掘;查询链

* Supported by the National Natural Science Foundation of China under Grant No.60603027 (国家自然科学基金); the Science-Technology Development Project of Tianjin of China under Grant No.04310941R (天津市科技发展计划); the Applied Basic Research Project of Tianjin of China under Grant No.05YFJMJC11700 (天津市应用基础研究计划)

Received 2006-05-03; Accepted 2007-02-05

中图法分类号: TP391 文献标识码: A

拼写校正是指针对由拼写检查器检测出存在于文本中的每个拼写错误,作出 1 个或多个更正建议的过程.通常情况下,拼写检查器会把未在给定词典里出现的字符串识别为错误拼写.本文解决了在英文文本中校正非词错误(比如将 the 拼成了 teh)的问题.我们不考虑将一个单词错拼成另一个单词的真词错误(比如将 form 拼成了 from).换句话说,我们暂不考虑上下文对单词拼写校正的影响,这主要基于目标应用、评测集合和系统性能的考虑.本文的目标是提出一个更好的可以应用到文本处理程序(比如 Microsoft Word 和 Aspell)中的拼写校正方法.在这些应用中,一定数量的(通常是 5~10 个)更正建议以一种交互的方式展现给用户,由用户选出最理想的一个.大多数情况下,排到第一位的建议是优先推荐的,因此,侧重点在 top-1 的准确度(top-1 accuracy,答案排在第一位的样本比例)上.此外,由于前 5/10 个建议也会展现给用户,所以我们同样关心 top-5/top-10 的准确度. Aspell[1]是目前比较流行的一个跨平台的拼写校正程序.文献[2]中的实验显示,Aspell 在其测试集上的 top-5 准确度超过了 85%,而 top-1 的准确度只有不到 60%.

我们的工作重点在于改进 Aspell 对候选排序的方法,以此获得更高的更正准确率.本文的主要贡献是:

1) 采用了一个判别式模型对 Aspell 的候选集合进行重排序,以一些额外的特征作为判据.由于这个模型(ranking SVM)具有较好的通用性,所以我们可以把现今最流行的多种拼写校正技术整合到这个模型中来,发挥它们各自的作用.这些技术包括编辑距离、基于字母的 *n* 元语法、发音相似度和噪音信道模型.在后面的实验中,这种模型展现了理想的性能;

2) 提出了在搜索引擎查询日志中自动获取拼写校正训练对的一种方法.这种方法抽取出的训练对质量也由实验结果加以验证.

第 1 节讨论研究人员在拼写校正领域所做的相关工作.第 2 节对重排序问题进行阐述.实验评估被安排在第 3 节,这一节同时会讲述从搜索引擎查询日志链中自动抽取训练对的细节.最后一节是总结和对本文模型的改进建议.

1 相关工作

比较全面的关于拼写校正的综述性文章主要有文献[3,4].几十年来,研究人员提出了很多算法来更正拼写错误.编辑距离和基于字母的 *n* 元语法可以用来处理排字错误(一般由键盘键位布局造成)[5],而 Soundex[4]和 metaphone[6]则适用于发音错误(比如将 phone 错拼成 fone).针对发音错误的算法都是将(正确或错误的)拼写映射到一个表示音节的编码串.比如,Soundex 用了 7 个码,Phonix 用了 9 个,而文献[7]中用的是 14 个.metaphone 及后续的 double metaphone[8]不仅考虑了字母的拼写,还考虑了发音因素.由于这些算法只能针对某类具体的错误,它们并不能完全胜任各种实际情况.于是,对这些算法的改良和采用多种算法的混合系统被开发出来.Phonix 对 Soundex 进行了改良,基于 Phonix 的系统[9,10]主要用于人名识别.Editex[11]综合考虑了 Soundex 和编辑距离算法.对于来自同一个 Soundex 字母组的编辑操作,它会赋给一个较低的惩罚项.Aspell 是一个开源的拼写程序,它采用了加权的编辑距离算法和 metaphone 算法.Aspell 拥有良好的性能,这主要是由于它综合考虑了较为先进的 metaphone 算法和 Ispell[12]的临近缺失策略:候选的生成操作可以是插入一个空格或连字符、交换两个临近的字母或替换、删除、添加一个字母.而本文提出的系统同时整合了多种拼写校正技术,而不仅仅依赖于一两个因素.

在拼写校正任务中用到的模型主要分为规则和统计两类.基于规则的研究包括文献[13,14].近些年来,统计机器学习方法也应用到这个任务中来,收到了良好的效果.统计方法大致可以分为两类:生成模型(generative model)和判别模型(discriminative model).在这个领域中广为应用的生成模型主要是噪音信道模型(noisy channel model),包括文献[2,15–17].噪音信道模型将错误拼写的生成看作是把一段文本向带有噪音的信道进行传输的过程,在传输过程中引起了拼写错误.对于每个可能的错误拼写,具有最大后验概率(即从输入拼写到该候选的转换概率)$P(candidate|input)$的候选作为建议返回.通过使用贝叶斯公式,并将作为常数的分母约去,可以

按照语言模型(或称为源模型)$P(candidate)$与信道模型(或称为纠错模型)$P(input|candidate)$的乘积作为分数对每个候选进行打分评判.语言模型可以用基于字母的 n 元语法来进行估计[15,18].而信道模型则通常归结为字母到字母[18]或字符串到字符串的混淆概率[15].在现今的主流英文拼写校正模型中,文献[15]由于其信道模型允许广义的字符串操作(如后缀 ant 变为 ent)具有最好的性能.对于判别式模型,Winnow[19]和神经网络[7]被用到了拼写校正任务中来.文献[19]中描述的模型对上下文敏感,采用了基于歧义消解的方法处理来自同一个混淆集(confusion set,比如 form 和 from)的用词错误.本文提出的模型是一个线性判别模型,它可以将各种影响因素以特征的形式整合进来,同时具有在训练过程中按照某种学习策略(如梯度下降、二次规划等)自动调整各个特征对应权值的特性.这些特性使得我们能够将现今流行的多种拼写校正技术放到同一个模型中去,并且取得了理想的效果.

除了在文本处理中的应用,日渐发展的搜索引擎技术也给查询拼写校正带来了极大的挑战,因为包含错误拼写的查询会带来网络带宽、计算资源和用户时间上的浪费.文献[16]中的统计数据表明,约有多达 10%~15%的查询包含一个或多个错误.不仅如此,查询中包含有大量的未登录词(out-of-vocabulary);而查询的平均长度小于 3 个、可以利用的上下文信息极为有限.这些都给查询拼写校正带来了很大的难度.现今关于搜索引擎上的拼写校正研究主要基于噪音信道模型[2,16].尽管本方法针对单个词的拼写错误,相信由于模型的通用性,经过适当的调整(比如增减特征),它也能很好地用于查询的拼写校正工作.

值得一提的是,中文拼写校正方面的开拓性工作可以参阅文献[29–31].由于汉字不是拼音文字,电子文档中汉字的拼写错误主要是由于音近(如“按步就班”中的“步”应为“部”)、形近(如“人”和“入”)、意近(如“既往不究”中的“究”应为“咎”)、输入法的键盘输入序列引起的[29].类似于文献[19],这些系统主要是针对混淆集中的单词在上下文搭配上引起的错误,采用歧义消解的思路进行校正.

2 问题描述

2.1 重排序问题定义

一个重排序问题可以作如下阐述:首先使用一个基准模型(baseline,是评判本方法性能好坏的标准)生成 N-best 候选(即选取基准模型打分排名在前 N 位的候选),附带上它们在基准模型中的得分(或者仅仅是先后顺序,比如本文).接下来使用一个新的模型,采用一些附加特征作为依据,对这些候选进行重新排序,让标准答案或者较接近答案的候选排名尽可能靠前,从而提高系统性能.形式化定义如下:

- 训练数据是一个输入(可能是错误拼写)/标准答案对的集合.在拼写校正任务中将训练样本表示为$\{q,a\}$.这里,q 表示为一个字符串(可能是错误拼写),而 a 则是该字符串对应的标准答案.
- 针对每个输入会依据某种规则生成若干候选,我们使用 c_{ij} 来表示第 i 个训练样本的第 j 个候选,$C(q_i)=\{c_{i1},c_{i2},...\}$表示输入 q_i 对应的候选集合.本文中 Aspell 生成的前 N 个候选被作为候选集合.
- $b(c,q)$是基准模型对输入 q 的候选 c 赋予的评分名次.这个信息将被作为重排序阶段的一个特征.若不使用该特征,则重排序模型(reranking model)就演变为一个排序模型(ranking model).
- 重排序模型中用到的额外特征(除基准系统排序特征之外的)用 $e_s(c,q)$来表示,$s=n+1,\ldots,m$.理论上讲,这些特征可以是候选的任意函数,最好是把那些有助于将好的候选同差的候选区分开来的特征包括进来.
- 模型的参数由一个 m 维向量来表示:$w=\{w_1,w_2,\ldots,w_m\}$.这个函数可以表示为

$$f(w,c,q)=\sum_{r=1}^{n} w_r b_r(c,q)+\sum_{s=n+1}^{m} w_s e_s(c,q) \tag{1}$$

这里,$b_1(c,q),\ldots,b_n(c,q)$表示基准模型的初始排序特征,而 $e_{n+1}(c,q),\ldots,e_m(c,q)$是用在重排序中的附加特征.将这些特征合起来表示为一个向量 $h(c,q)=\{b_1(c,q),\ldots,b_n(c,q),e_{n+1}(c,q),\ldots,e_m(c,q)\}$,那么,候选排序的结果是

$$f(w,c,q)=w\cdot h(c,q).$$

接下来的机器学习任务就是寻找一组在测试集上能够展现良好性能的参数向量 w.这个过程是通过在训练样本上的训练来完成的.下面讨论如何使用 Ranking SVM 来学习这个向量.

2.2 Ranking SVM

排序问题是指学习如何给一组对象按照一定标准设定它们之间的相对顺序,它是在近年来的机器学习研究中一个很受关注的问题[20−22].不同于传统的机器学习任务——分类和回归,Ranking SVM[20]被定义为将不同对象映射到某种序关系上.在人们的偏好关系起重要作用的一些领域,如社会科学和信息检索,排序问题十分普遍.这里使用的 Ranking SVM 可以被看作是对序数回归(ordinal regression)SVM 的推广[23].

使用 Ranking SVM 的一个关键是如何对用户的偏好判断进行建模,从而导出排序的约束关系.文献[20]中使用了点击日志(clickthrough)来确定用户偏好,而在文献[21]中则采用了查询链的概念.根据偏好判断可以得出损失函数使其最小化.若存在一个针对输入 q 的候选 c_i 和 c_j 上的喜好判断,它可以被表述为如下形式:

$$c_i >_q c_j \tag{2}$$

以上的判定规则表明:给定输入 q,候选 c_i 比 c_j 的优先级更高一些.每个训练样本通常包含多个这样的约束.对于 Ranking SVM,可以将式(2)转述为(特征函数族 h 将候选映射为一个多维特征向量):

$$w\cdot h(c_i,q)>w\cdot h(c_j,q) \tag{3}$$

对于每一个输入的训练对,上述约束可以在传统的分类 SVM 表达为(同时去掉一般线性判别模型表示式 $w\cdot h+b$ 中的偏置 b,移项,可得式(4),这相当于分类中一个 $h_i–h_j$ 实例):

$$w\cdot[h(c_i,q)–h(c_j,q)]>0 \tag{4}$$

加上一个间隔(margin)和非负的松弛变量,从而允许某些偏好约束被违反,由此可以最小化它们的上界 $\sum\xi_{ij}$.这样,就将最大化间隔的问题转化为如下的一个凸二次规划问题(其中,C 是在训练误差和间隔的选取折衷的一个参数):

$$\begin{aligned}&\min_{w,\xi_{ij}} \frac{1}{2}w\cdot w + C\sum_{ij}\xi_{ij}\\ &\text{subject to}\\ &\forall q,i,j: w\cdot h(c_i,q) >_q w\cdot h(c_j,q)+1-\xi_{ij}\\ &\forall i,j:\xi_{ij}\geq 0\end{aligned} \tag{5}$$

2.3 拼写校正的判别式重排序

本文根据上述的重排序框架,采用 Ranking SVM 来学习目标参数集合.这个重排序框架结构上类似于文献[24,25].但由于基准模型 Aspell 对每个候选的打分难以获取,只能将候选的相对排名 $R(c_{ij})$作为二进制特征(比如候选在基准模型的排序中是否在前 N 位,N=1,3,5,…),而不是一个实值特征.

本任务的目标是对每个拼写错误提出 5~10 个建议,让用户选择最理想的建议,因此,不仅要将答案排到第一的位置,同时也需要把那些接近答案的候选(比如与答案具有相同的词根)的排序位置尽可能提前.而当用户的输入被识别为一个错误拼写时,通常他们并不十分确定应该选择系统给出的哪个拼写建议.在这种情况下,本文使用一定的策略来模拟用户的喜好判断,然后给每个样本中的不同候选赋予不同的排序等级.对于本任务,一个显而易见的策略是将标准答案放在第 1 等级,其变形(inflection,如复数形式、过去式、过去分词等)作为第 2 等级,派生词作为第 3 等级,其他词为第 4 等级.由于常见的判别算法(如 Perceptron、最大熵)不能将每个样本中的多个排序约束有效表达出来,而 Ranking SVM 能够很好地做到这一点,最终选取 Ranking SVM 用作训练.

2.4 特征模板

Ranking SVM 可以把目前比较成熟的拼写校正技术整合到一个统一的判别式模型中.这些技术包括编辑距离、发音相似度、基于字母的 n 元语法和噪音信道模型.用到的特征模板见表 1.

如第 1 节所述,噪音信道模型的得分由语言模型估计值与信道模型估计值的乘积得出.文中用到的依赖上下文的加权编辑距离按照文献[15]的方法计算得出.而基于字母的 n 元语法相似度计算方法是文献[26]中公式的一个变形.模板中用到的 double metaphone 算法[8]是 metaphone 算法[6]的改进版本.

Table 1 Feature templates

表 1 特征模板

Category	Feature	Description
Aspell' s initial ranking	*C_IsBaselineRankTop⟨N⟩*	Ranked top-*N* from Aspell candidate list (*N*=1,5,10,…)
Noisy channel score	*C_NoisyChannelScoreTop⟨N⟩*	Ranked top-*N* with noisy channel score (*N*=1,5,10,…)
Frequency	*C_UniFreqRatio_GT_⟨N⟩*	The unigram freq. ratio of candidate and input >*N*?
	C_UniFreq_GT_⟨N⟩	The unigram freq of candidate >*N*?
Edit distance	*C_EditDist_LE_⟨N⟩*	Edit distance *b*/*w* input & candidate <=*N*?
	C_EditDist_GE_⟨N⟩	Edit distance *b*/*w* input & candidate >=*N*?
Lexicon	*C_QryinLogLex_&_Namelist*	Is input in log lexicon or name list?
	C_inTrustLex_&_Namelist	Is candidate in log lexicon or name list?
Phonetic	*C_DMPKeyEQ2Input*	Do the candidate & input share the same double metaphone key?
Length	*C_Len_LE_⟨N⟩*	The length of the candidate <=*N*?
	C_Len_GE_⟨N⟩	The length of the candidate >=*N*?
	C_LenChanged	Length(input) != length(candidate)?
Letter-Based *n*-gram	*C_NgramSimilar2Input*	Is Similarity$_{n\text{-gram}}$(candidate, input) >=*N* (*N* ranges from {0, 1})

3 实验结果

3.1 训练数据(查询链训练集和人工标注训练集)

一个判别式学习任务需要一些(输入,标准答案)对来进行训练.通常这些训练对的标注是由人工完成的,工作枯燥,耗时耗力.本文探索出了一条能够在搜索引擎查询日志中自动抽取训练对的途径,大量减少了人力物力需求.这种方法采用了文献[21]提出的查询链概念.该文指出,用户为满足某个信息需求而搜索网页的过程,往往会对查询进行不断修改而形成一个查询序列.他们把这种序列称作查询链,并将这个概念用于搜索引擎排序策略的改进上.

本文将这个概念应用到从查询日志中抽取拼写校正训练对.如图 1 所示,假设某个用户想要获得 messenger 软件最新版本的下载链接,他向搜索引擎提交查询"messenger download".搜索引擎在返回结果的同时,给用户提出了一个拼写建议"messenger download".一开始,该用户可能感到困惑,后来,他可以根据外界的种种信息发现查询中的拼写错误并加以改正.用户重新提交查询后便会得到想要的结果.通过一系列的日志挖掘步骤,我们得到训练对(messenger download,messenger download).而训练对(resipi,recipe)也可以按类似方式抽取出来.由于只考虑单个词的拼写校正,对以上两个对子我们只保留(messenger,messenger)和(resipi,recipe).

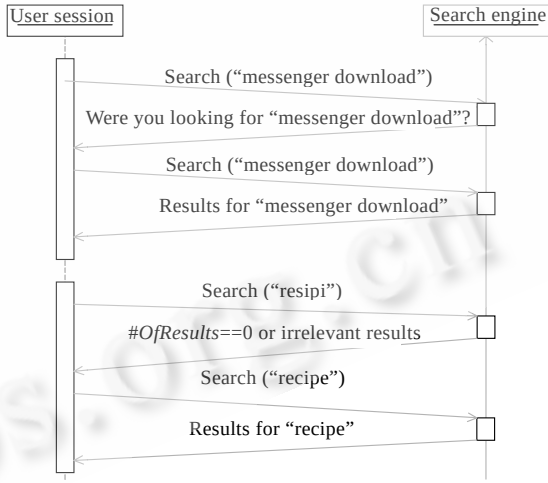

Fig.1 Query chain

图 1 查询链

本文认为,查询链实质上是人认知过程在查询日志中的记录.如果搜索引擎返回的结果不令人满意(结果不相关、网页数目很少甚至没有),人们会用自己的知识或者借助外部资源(查阅词典、点击搜索引擎上的建议链接查看新的搜索结果,等等)对查询进行修正.这个过程不断往复,便在查询日志中留下了一个查询序列.当然,用户中途放弃的情况也应当是常见的.后继的实验表明,这种情况对训练对抽取质量的影响很小,基于查询链训练对的模型获得了与基于人工标注训练对模型相近的性能.

我们从 MSN Search[27]持续 5 个月的查询日志中采样了 5 天的数据,共计 120 个文件,约 1 500 万个英文查询记录.一小时内同一个 IP 地址的查询请求被界定为一个用户会话,然后用一些启发式(如基于字母的 *n* 元语法、编辑距离等)对这些查询进行分组.将每个组内的查询按时间戳排序,假定后面输入的查询要比前面的更接

近标准答案.抽取出的查询对满足:1) 错词(如 resipi)不在词典中,而更正词(如 recipe)在词典中;2) 错词与更正词的编辑距离与词的长度成正比.本文从 24 288 个抽取的查询链训练对中随机采样了 1 000 个用作本次实验的一个训练集合.为了验证查询链训练对的有效性,又使用了 1 000 个人工标注训练对作为第 2 个训练集合.

3.2 测试数据

本文使用 Aspell 公布的测试集(http://aspell.net/test/)作为测试数据.它包括 547 个不同的人工标注样本(形如⟨theirselves,themselves⟩,⟨wicken,weaken⟩).为了与文献[2]中提出的 EMBED 模型进行比较,按照该文做法,从测试集中去除复合词,留下 508 个样本.我们目前的方法忽略了只有大小写区别的拼写错误.

3.3 评估指标

采用精度(precision)、召回率(recall)和 top-*N* 准确度(accuracy)作为评估指标.这里定义的精度针对那些作出修改(给出的最优建议不同于输入)的样本,而召回率与 top-1 准确度等价.计算公式如下:

$$\text{精度}(Precision)=\frac{\text{改正确样本数}}{\text{改正确样本数}+\text{改错误样本数}} \tag{6}$$

$$\text{召回率}(Recall)=\frac{\text{改正确样本数}}{\text{改正确样本数}+\text{未作修改样本数}+\text{改错误样本数}} \tag{7}$$

$$Top-N\text{准确充}(Accuracy_{TopN})=\frac{\text{答案作为候选排名在前}N\text{位的样本数}}{\text{样本总数}} \tag{8}$$

$$N=1,5,10,25,100$$

3.4 基于Aspell初始排序的结果(重排序模型)

本文使用 Aspell 的最新版本(0.60.4)生成 *N*-best 候选,这些候选根据 Aspell 的评分从大到小排序.Aspell 有 5 种建议模式:*ultra*,*fast*,*normal*,*slow* 和 *bad-spellers*.*slow* 模式输出的候选具有较好的覆盖率(答案在候选中的样本比例)和适当的候选集大小(每个样本不超过 100 个候选),本实验选取了这种模式.对于 Ranking SVM 使用了 SVM^{light} v6.01[28],包括损失函数在内的所有参数都使用了默认值.我们使用了一种贪婪策略进行特征选择,最终按照第 2.4 节的特征模板选定了 16 个特征.单词词频和词典统计数据来自 MSN Search[27]的 9 个月的统计数据.BBC(英国广播公司)提供的 Perl 模块 Lingua::MSWordSpell(http://search.cpan.org/~bbc/Lingua-MSWordSpell-1.010/lib/Lingua/MSWordSpell.pm)被用来对 Microsoft Word 2003 的 top1/5/10 准确度进行评估.采用同样的候选集合,我们重新实现了文献[15]中描述的信道模型,最大窗口的大小优化为 3.*n* 元语法被用来(在本任务下转变为一元模型)对源模型 *P*(*candidate*)进行估计.

表 2 将本方法(ranking SVM)与一些拼写校正系统进行了比较(N/A 表示这些系统没有提供这项指标).本方法在 top 1/5/10 的准确度上比基准系统有了明显的提高.在这个测试集上,基于查询链训练集的结果与人工标注训练集的结果相差很小,它们针对基准系统的错误降低率分别达到了 32.2%和 32.6%.同时,本系统也超过了文献中表现最好的模型[15]和商用系统(如 Microsoft Word 2003)拼写校正模块.我们认为,这一方面是利用了 Aspell 初始排序信息,另一方面则引入了包括噪音信道模型在内的一些特征.由于用到的特征较多,计算复杂度相对大一些,这也是今后改进和优化的方向之一.

Table 2 Top-*N* accuracy

表 2 Top-*N* 准确度

	EMBED[2]	Microsoft Word 2003	Brill & Moore's error model[15]	Aspell 0.60.4 (baseline)	Ranking SVM (query chain)	Ranking SVM (human annotated)
# Samples	508	508	508	508	508	508
Top 1	211 (41.5%)	306 (60.2%)	312 (61.4%)	269 (53.0%)	346 (68.1%)	347 (68.3%)
Top 5	331 (65.2%)	377 (74.2%)	429 (84.4%)	410 (80.7%)	434 (85.4%)	440 (86.6%)
Top 10	N/A	382 (75.2%)	445 (87.6%)	443 (87.2%)	448 (88.2%)	452 (89.0%)
Top 25	386 (76.0%)	N/A	460 (90.6%)	456 (89.8%)	460 (90.6%)	461 (90.7%)
Top 100	402 (79.1%)	N/A	462 (90.9%)	462 (90.9%)	462 (90.9%)	462 (90.9%)

3.5 不基于Aspell初始排序的结果(排序模型)

本文所提出的方法涉及的一个问题是:如果不加入 Aspell 初始排序信息,单独使用那些来自多种拼写校正技术的特征,由此训练出来的系统性能如何?由于不使用基准模型的排序信息,重排序模型(reranking model)演变成为一个排序模型(ranking model).

为了验证这些特征的有效性,我们进行了两次后继实验(查询链和人工标注训练集合不变).不使用 Aspell 的初始排序信息,对 Aspell 的 *N*-best 候选进行排序.这两次实验的结果见表 3.

Table 3 Performance of the ranking and reranking model

表 3 排序模型和重排序模型的性能比较

	Aspell 0.60.4 (baseline)	Ranking SVM (query chain, ranking)	Ranking SVM (human annotated, ranking)	Ranking SVM (query chain, reranking)	Ranking SVM (human annotated, reranking)
# Samples	508	508	508	508	508
Precision (%)	53.3	64.4	65.4	70.3	70.5
Reduced error rate (%)	0.0	20.1	22.2	32.2	32.6
Top 1 accuracy (recall)	269 (53.0%)	317 (62.4%)	322 (63.4%)	346 (68.1%)	347 (68.3%)
Top 5 accuracy	410 (80.7%)	443 (87.2%)	440 (86.6%)	434 (85.4%)	440 (86.6%)
Top 10 accuracy	443 (87.2%)	452 (89.0%)	456 (89.8%)	448 (88.2%)	452 (89.0%)
Top 25 accuracy	456 (89.8%)	461 (90.7%)	462 (90.9%)	460 (90.6%)	461 (90.7%)
Top100 accuracy	462 (90.9%)	462 (90.9%)	462 (90.9%)	462 (90.9%)	462 (90.9%)

可以清楚地看到,在不使用基准系统排序信息的前提下,本判别模型给出了令人信服的结果.其中,查询链排序模型将错误率降低了 20.1%,而人工标注排序模型降低了 22.2%.尽管如此,排序模型与重排序模型仍存在约 5%的差距.本文认为,这应归结于 Aspell 良好的打分策略,它使用了一系列复杂的启发式规则.

4 结论和改进方向

本文提出了一种基于判别式模型 Ranking SVM 的拼写校正方法.该方法对基准拼写校正系统 Aspell 的输出进行重排序,显著地提高了基准系统 top 1/5/10 的准确度.由于该判别模型的通用性,得以将现今较为成熟的多种拼写校正技术(编辑距离、发音相似度、基于字母的 *n* 元语法和噪音信道模型)整合进来.与人工标注训练集相比,由查询链抽取的训练对训练出的模型也取得了很好的结果.

尽管如此,这一工作仍然存在需要改进之处:首先,可以对拼写校正之前的拼写检查模块进行扩展,将错误拼写的上下文考虑进来.我们认为,这种做法能够更好地适应实际应用;另一方面,还可以寻找更多的具有更高区分度、更低时间空间复杂度的特征.例如,用户点击搜索引擎给出的拼写建议链接,产生的点击记录(clickthrough)可以作为一个重要的启发式.由于搜索引擎已经成为人们访问海量信息的入口,我们可以更进一步地分析查询中的错误拼写及其改正答案在返回网页摘要(snippets,甚至这些摘要的来源网页)的分布.此外,查询链的作用还可以进一步探索,比如用于查询扩展.

致谢 在此,我们向对本文的工作给予支持和建议的同行,尤其是上海交通大学的包胜华、袁伟以及重庆大学的陈议同志表示感谢.

References:

[1] Aspell. 2007. http://aspell.net
[2] Ahmad F, Kondrak G. Learning a spelling error model from search query logs. In: Proc. of the EMNLP 2005. 2005. 955−962.
[3] Jurafsky D, Martin J. Speech and Language Processing. Prentice Hall. 2000.
[4] Kukich K. Techniques for automatically correcting words in text. ACM Computing Survey, 1992,14(4):377−439.
[5] Damerau FJ. A technique for computer detection and correction of spelling errors. Communications of the ACM, 1964. 171−176.
[6] Philips L. Hanging on the metaphone. Computer Language Magazine,1990,7(12):38−44.
[7] Hodge VJ, Austin J. A comparison of standard spell checking algorithms and a novel binary neural approach. IEEE Trans. on Knowledge and Data Engineering, 2003,15(5):1073−1081.
[8] Philips L. The double-metaphone search algorithm. C/C++ User's Journal, 2000,18(6). http://www.cuj.com/documents/s=8038/cuj0006philips/
[9] Erikson K. Approximate swedish name matching—Survey and test of different algorithms. NADA Report, TRITA-NA-E9721, 1997.

[10] Gadd T. PHONIX: The algorithm. Program, 1990,24(4):363−366.
[11] Zobel J, Dart P. Phonetic string matching: Lessons from information retrieval. In: Proc. of the ACM SIGIR. 1996. 166−172.
[12] Ispell. http://www.gnu.org/software/ispell/ispell.html
[13] Mangu L, Brill E. Automatic rule acquisition for spelling correction. In: Proc. of the 14th ICML. 1997. 187−194. http://citeseer.ist.psu.edu/mangu97automatic.html
[14] Martins B, Silva MJ. Spelling correction for search engine queries. In: Proc. of the EsTAL. 2004. 372−383. http://www.springerlink.com/content/m7g3d9tt351urg7f/
[15] Brill E, Moore RC. An improved error model for noisy channel spelling correction. In: Proc. of the 38th Annual Meeting of Association for Computational Linguistics. 2000. 286−293.
[16] Cucerzan S, Brill E. Spelling correction as an iterative process that exploits the collective knowledge of Web users. In: Proc. of the EMNLP. 2004. 293−300. http://citeseer.ist.psu.edu/754653.html
[17] Toutanova K, Moore RC. Pronunciation modeling for improved spelling correction. In: Proc. of the 40th Annual Meeting of Association for Computational Linguistics. 2002. 144−151. http://citeseer.ist.psu.edu/541572.html
[18] Kernighan MD, Church KW, Gale WA. A spelling correction program based on noisy channel model. In: Proc. of the 13th COLING, Vol. 2. 1990. 205−210. http://citeseer.ist.psu.edu/kemighan90spelling.html
[19] Golding AR, Roth D. Applying winnow to context-sensitive spelling correction. In: Proc. of the 13th ICML. 1996. 182−190. http://citeseerist.psu.edu/golding96applying.html
[20] Joachims T. Optimizing search engines using clickthrough data. In: Proc. of the 8th ACM SIGKDD. 2002. 133−142. http://portal.acm.org/citation.cfm?id=775067
[21] Radlinski F, Joachims T. Query chains: Learning to rank from implicit feedback. In: Proc. of the 11th ACM SIGKDD. 2005. 239−248. http://citeseer.ist.psu.edu/radlinski05query.html
[22] Xu J, Cao YB, Li H, Zhao M. Ranking definitions with supervised learning methods. In: Proc. of the WWW. 2005. 811−819. http://portalacm.org/citation.cfm?id=1062745.1062761
[23] Herbrich R, Graepel T, Obermayer K. Large margin rank boundaries for ordinal regression. In: Smola A, Bartlett P, Schölkopf B, Schuurmans D, eds. Advances in Large Margin Classifiers. 2000. 115−132.
[24] Collins M. Discriminative reranking for natural language parsing. In: Proc. of the 17th ICML. 2000. 175−182. http://citeseer.ist.psu.edu/collins00discriminative.html
[25] Collins M. Ranking algorithms for named-entity extraction: Boosting and the voted perceptron. In: Proc. of the 40th Annual Meeting on Association for Computational Linguistics. 2002. 489−496. http://citeseer.ist.psu.edu/collins02ranking.html
[26] Lin DK. An information-theoretic definition of similarity. In: Proc. of the 15th ICML. 1998. 296−304. http://portal.acm.org/citation.cfm?id=657297
[27] MSN Search. http://search.msn.com
[28] Joachims T. Making large-scale SVM learning practical. In: Schölkopf B, Burges C, Smola A, eds. Advances in Kernel Methods—Support Vector Machines. MIT Press, 1999.
[29] Chang CH. A new approach for automatic Chinese spelling correction. COLIPS, 1994,4(2):143−149.
[30] Li JH, Wang XL. Combine trigram and automatic weight distribution in Chinese spelling error correction. Journal of Computer Science Technology, 2002,17(6):915−923.
[31] Zhang L, Zhou M, Huang CN, Lu MY. Approach in automatic detection and correction of errors in Chinese text based on feature and learning. In: Proc. of the 3rd Chinese World Congress on Intelligent Control and Intelligent Automation, Vol.4. 2000. 2744−2748 (in Chinese with English abstract). http://ieeexplore.ieee.org/iel5/6941/18672/00862557.pdf?arnumber=862557

附中文参考文献:

[31] 张磊,周明,黄昌宁,鲁明羽.基于特征与学习的中文文本自动校对方法.见:第3届中文智能控制及智能自动化会议论文集,第4卷. 2000. 2744−2748. http://ieeexplore.ieee.org/iel5/6941/18672/00862557.pdf?arnumber=862557

张扬(1981－),男,重庆人,硕士生,主要研究领域为机器学习,自然语言处理,信息检索.

向伟(1984－),男,博士生,主要研究领域为知识发现,数据挖掘.

何丕廉(1942－),男,教授,博士生导师,主要研究领域为自然语言处理.

李沐(1972－),男,博士,研究员,主要研究领域为自然语言处理.