基于位置语言模型的中文信息检索系统的研究

陈雅兰[1] 胡小华[1,2] 涂新辉[1] 何婷婷[1]
(华中师范大学计算机学院 武汉 430079)[1] (德雷塞尔大学信息科学与技术学院 费城 19082)[2]

摘 要 在大多数现有的检索模型中常常忽略了如下事实:一个文档中匹配到的查询词项的近邻性和打分时所基于的段落检索也可以被用来促进文档的打分。受此启发,提出了基于位置语言模型的中文信息检索系统,首先通过定义位置传播数的概念,为每个位置单独地建立语言模型;然后通过引入KL-divergence检索模型,并结合位置语言模型给每个位置单独打分;最后由多参数打分策略得到文档的最终得分。实验中还重点比较了基于词表和基于二元两种中文索引方法在位置语言模型中的检索效果。在标准NTCIR5、NTCIR6测试集上的实验结果表明,该检索方法在两种索引方式上都显著改善了中文检索系统的性能,并且优于向量空间模型、BM25概率模型、统计语言模型。

关键词 位置语言模型,近邻性,段落检索,传播数
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.7.057

Positional Language Model-based Chinese IR System

CHEN Ya-lan[1] HU Xiao-hua[1,2] TU Xin-hui[1] HE Ting-ting[1]
(School of Computer Science, Central China Normal University, Wuhan 430079, China)[1]
(College of Information Science and Technology, Drexel University, Philadelphia 19082, USA)[2]

Abstract In most existing retrieval models, the facts are often overlooked that the proximity of matched query terms in a document and passage retrieval used to score can also be exploited to promote scoring for documents. Inspired by this, a Chinese information retrieval system based on the positional language model was proposed. Firstly, we defined the concept of propagated count to establish a positional language model for each position. Then through combing KL-divergence retrieval model and positional language model, we scored for each individual position. Finally, we scored the document by the multi-parameter strategy. The experiment also focuses on comparing the retrieval effect of the two Chinese indexing approaches named multi character-based and dictionary-based on positional language models. Experiments on standard NTCIR5, NTCIR6 test sets show that the performance of the two indexing approaches of IR system improves greatly and it performs better than the vector space model, okapi bm25 model and classical language model.

Keywords Positional language model, Proximity, Passage retrieval, Propagated count

1 引言

Ponte和Croft[1]在1998年首次提出基于语言模型的信息检索系统,从此新一代概率检索模型——语言建模方法被引入到信息检索中。近些年,许多改进的语言模型也相继被提出,但是它们大多数都是集中在提高查询语言模型[2,3]和文档语言模型[4]的估计上。这些语言模型虽然相对于向量空间模型等传统模型来说有着不一样的出发点,但是都趋向于在查询功能的实现方面做相应的改进。由于基于良好的统计基础,这些语言模型使设置和优化检索参数变得更加容易,而且检索效果往往优于传统的检索模型。

虽然过往的研究在这些语言模型上做了大量的工作,也取得了不错的效果,但是仍然有两个检索相关的语言建模方法没有被考虑到:(1)词项近邻性,使文档获得较高得分的查询词项在文档中出现的位置往往较为接近;(2)段落检索,在给文档打分时,主要依据的是文档中的最佳匹配段落。也就是说对于同一查询语句,当查询中的每个词项在两篇不同的文档中的出现频率一样时,各查询词项在文档中出现位置较靠近的文档理应获得更高的检索得分,但是统计语言模型无法做到这一点。

因此,Lv等[5]提出位置语言模型,该模型将词项近邻性和段落检索两个检索关键点统一到一个语言模型中。其关键思想就是为文档中的每一个词项位置建立一个语言模型,然后根据这些位置语言模型来给整个文档打分。但现有的研究

到稿日期:2014-08-01 返修日期:2014-10-27 本文受国家社会科学基金重大项目(12&2D223),湖北省自然科学基金重点项目(2011CDA034),国家语委"十二五"重点项目(ZDI125-1),国家"十二五"科技支撑计划课题(2012BAK24B01),教育部/国家外国专家局高等学校学科创新引智计划项目(B07042),华中师范大学中央高校基本科研业务费项目(CCNU13A05014,CCNU13C01001,CCNU13F010),国家自然科学基金(61300144)资助。

陈雅兰(1988-),女,硕士生,主要研究方向为信息检索,E-mail:furongyalan@163.com;胡小华(1965-),男,博士,教授,主要研究方向为数据挖掘、生物信息学;涂新辉(1979-),男,博士,副教授,主要研究方向为信息检索;何婷婷(1964-),女,博士,教授,主要研究方向为自然语言处理、数据库。

工作通常都是为整篇文档定义一个文档语言模型，这就形成了鲜明的对比。为每个位置引入一个语言模型，最大的优点就是它能方便我们利用普通的概率模型对文档中的“最佳匹配位置”进行建模，以获得更好的检索结果。

基于位置语言模型良好的检索性能，余伟[6]在该模型上融入词与词间的语义关系，提出结合语义的位置语言模型。Jun Miao[7]将位置语言模型中的词项近邻性应用到Rocchio's相关反馈模型中，通过考虑候选扩展词项与查询词项间的近邻性信息来提高查询准确率。Yuanhua Lv[8]把词项位置关系引入到反馈文档中来帮助选择查询主题词，提出了位置相关模型。过往基于位置语言模型的研究都取得了不错的成绩，因此本文欲将其应用到中文信息检索中，以探究其在中文文本上的检索性能。

但是中文与西方的拼音文字有很大的不同，在中文文本句子中，词和词之间并没有空格隔开。因为两种语言本质上的差异，适用于西方拼音文字的检索方法并不能直接应用到中文中。因此，中文文本索引的建立对检索系统性能的提高具有至关重要的作用。Kwok[9]、Lam[10]和Nie[11]比较了使用不同的中文索引技术时系统的性能。通常，基于单字的索引模型具有最好的召回率，而基于词或二元的索引则具有较好的准确率。与基于词的索引不同的是，二元索引不会受到未登录词的困扰，但是却需要消耗大量的存储空间。鉴于下面几个原因，并不能得出词索引比二元索引更优的结论：(1)分词词表不可能完全覆盖测试文档集合中所有的词；(2)不同的人、不同语言学家和不同的标准对词的定义不一样；(3)在实验中，二元索引和词索引往往获得相近的准确率。因此，本文还将重点比较词表和二元两种索引方法在位置语言模型上的应用效果。

本文旨在利用位置语言模型的建模优势，将其引入到中文信息检索中，通过不同索引方法应用及不同模型检索效果间的比较，检验位置语言模型在中文信息检索中的应用效果。实验结果表明，相对于传统的语言模型，该方法明显地改善了系统的检索性能。

2 基于位置语言模型的中文信息检索系统

首先介绍位置语言模型的基本原理与思路，然后介绍该系统的两个关键组成部分：基于近邻性的传播数的度量以及基于位置语言模型的文档排序策略，最后介绍基于位置语言模型的中文信息检索系统。

2.1 基本原理

在现有的针对语言模型的研究工作中，文档语言模型都是仅仅基于词项在文档中出现的数量来建立的，而与词项在文档中出现的位置无关。位置语言模型就打破了这一常规限制，即基于每个词项依赖于不同位置所出现的次数来估计语言模型。语言模型建立的思想就是对文档中每个词项的位置进行建模，把每个位置都看作是一个“隐形”的段落，而段落中的所有词就是整个文档中出现过的词项，只是出现的次数是根据离该位置的远近程度来度量的。

位置语言模型的建立基于以下两个假设：(1)在同一文档中，每个位置出现的词项能将其在该位置的出现信息通过基于近邻性的密度函数传播给其他的位置，这里称之为词项传播数。这个假设的想法就是，如果一个词项 w 出现在位置 i 处，那么就假设这个词项会以一定折扣的“出现次数”出现在文档的所有其他位置，这样在越靠近当前位置 i 的位置，词项 w 的“出现次数”(或称传播数)就越大，当然词项 w 到其他所有位置的传播数都会少于在当前位置 i 处的计数。(2)对于文档中的每个位置 i，都存在一个虚拟文档，这个虚拟文档包括原文档中的所有词项，位置语言模型根据各词项在位置 i 处的传播总数来建立，一旦模型建立，就按照一般的语言模型来匹配查询模型。

在建立位置语言模型之前，用 $D=(w_1,\cdots,w_i,\cdots w_j,\cdots,w_N)$ 来表示一篇文档，其中 $1,i,j,N$ 表示对应词项在文档中的绝对位置，N 表示文档中词的总数。而建模的关键点在于词项传播数的定义，首先，用 $c(w,i)$ 来表示词项 w 在位置 i 处出现的次数，若出现则为1，不出现则为0；$k(i,j)$ 表示位置 j 处的词项向位置 i 的传播数；$c'(w,i)$ 表示位置 i 处的词项 w 来自文档中其他所有出现该词项 w 的位置的传播总数，用公式可表示为：

$$c'(w,i)=\sum_{j=1}^{N}c(w,j)\cdot k(i,j) \tag{1}$$

因此，即使 $c(w,i)$ 为0，传播总数 $c'(w,i)$ 也一定大于0。如图1所示，经传播之后，位置 i 处词项 $q1$ 和 $q2$ 的计数都大于0。基于上述词项的传播数，在位置 i 处建立一个词频向量 $(c'(w_1,i),\cdots,c'(w_N,i))$，由该词频向量来形成位置 i 处的虚拟文档 D_i。由此可见，词项的位置信息被转换成了词项的词频信息存储在这个向量中。因此位置 i 处虚拟文档的语言模型可以定义为：

$$p(w|D,i)=\frac{c'(w,i)}{\sum_{w'\in V}c'(w',i)}=\sum_{j=1}^{N}\frac{k(i,j)}{\sum_{j'=1}^{N}k(i,j')}\cdot c(w,j) \tag{2}$$

其中，V 表示整个词汇集合。把 $p(w|D,i)$ 叫做位置 i 处的位置语言模型。

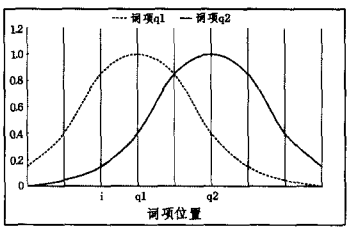

图1 词项传播实例

对于一个给定的查询 Q，使用 w 在查询 Q 的分布与在文档 D 中位置 i 的分布的差异性(负KL散度)来衡量文档 D 中位置 i 处的检索得分：

$$S(Q,D,i)=-\sum_{w\in V}p(w|Q)\cdot\log\frac{p(w|Q)}{p(w|D,i)} \tag{3}$$

其中，$p(w|Q)$ 是一个估计的查询语言模型，可以用最大似然估计或是一些伪相关反馈算法来估计，例如相关模型[3]或混合模型[2]。

类似于常规的语言模型，位置语言模型也需要用一些平滑方法来解决零概率问题和惩罚一些常用术语[12]。模型考虑了Dirichlet Prior和Jelinek-Mercer两种平滑方法，Dirichlet Prior已经被证明是一种较有效的文档语言模型平滑方法，并且能捕获文档的长度特征[12]。位置语言模型采用一般的集合语言模型 $p(w|C)$ 作为背景模型，故平滑模型可以定义为：

$$p_\mu(w|D,i)=\frac{c'(w,i)+\mu p(w|C)}{\sum_{w\in V}c'(w,i)+\mu} \tag{4}$$

其中，μ 是平滑参数。尽管先前的一些工作都已明确显示Je-

linek-Mercer 平滑方法没有 Dirichlet Prior 方法的检索效果好[12]，但由于本模型文档中不同位置的虚拟文档长度是相似的，并不能判断这个结论在位置语言模型中是否成立，所以在实验中也会考虑 Jelinek-Mercer 平滑方法。平滑方法如下：

$$p_\lambda(w|D,i)=(1-\lambda)p(w|D,i)+\lambda p(w|C) \tag{5}$$

其中，λ 是平滑参数。

2.2 传播数度量

很显然，建立位置语言模型最主要的技术挑战就是如何定义这个传播数函数 $k(i,j)$。基于其他研究者所做的一些工作[13-15]，在这里考虑 4 种有代表性的核函数：高斯、三角、余弦、和圆，不同核函数会导致不同的位置语言模型。

(1)高斯核函数(Gaussian)

$$k(i,j)=e^{\frac{-(i-j)^2}{2\sigma^2}} \tag{6}$$

(2)三角核函数(Triangle)

$$k(i,j)=(1-\frac{|i-j|}{\sigma})\cdot 1_{\{|i-j|\leqslant\sigma\}} \tag{7}$$

(3)余弦核函数(Cosine)

$$k(i,j)=\frac{1}{2}[1+\cos(\frac{|i-j|\cdot\pi}{\sigma})]\cdot 1_{\{|i-j|\leqslant\sigma\}} \tag{8}$$

(4)圆核函数(Circle)

$$k(i,j)=\sqrt{1-(\frac{|i-j|}{\sigma})^2}\cdot 1_{\{|i-j|\leqslant\sigma\}} \tag{9}$$

可以看到，这 4 个核函数都有一个共同的参数 σ，这个参数用来控制内核曲线的伸展，这就是说它会限制每个词项的传播范围。一般来说，对于不同的词项甚至是不同的查询来说，参数 σ 的最优设置应该是不同的，这是因为在一篇文档中一些通用术语会拥有比较广的语义范围，因此需要一个较高的 σ 值。类似地，一些常用的查询词项也会比那些更为具体的查询词项匹配到更长的相关段落。但是这里，对所有的词项和查询只是简单地把 σ 设置为固定的值。

2.3 文档排序策略

如前文所述，通过当前位置 i 处的位置语言模型和查询语言模型之间的负 KL 散度可以为文档中的每个位置 i 计算一个具体的位置得分 $S(Q,D,i)$，然后通过这些具体的位置得分来计算文档 D 的总得分。本文采用多参数 σ 打分策略。

在这个打分策略中，先对几个不同的 σ 值分别计算其对应的最佳位置得分，然后合并这些得分来作为文档的最后得分。该打分策略的思想是通过用不同的 σ 值来捕获不同传播范围的近邻性。

$$S(Q,D)=\sum_{\sigma\in R}[\beta_\sigma\cdot\max\{S_\sigma(Q,D,i)\}] \tag{10}$$

其中，R 是预先定义好的 σ 值集合，$S_\sigma(\cdot)$ 是位置语言模型对应特定 σ 值的得分函数，β_σ 表示不同 σ 参数值的权重。特别地，当 $R=\{\sigma_0,\infty\}$ 时，该打分策略就等价于带参数 σ_0 的位置语言模型和常规的文档语言模型的插值。考虑到效率问题，只估计该多参数 σ 策略的特殊情况，定义如下：

$$S(Q,D)=\gamma\max_{i\in[1,N]}\{S_{\sigma_0}(Q,D,i)\}+(1-\gamma)[-D(\theta_Q\|\theta_D)] \tag{11}$$

2.4 中文信息检索系统的实现

结合位置语言模型的建模优势，将其引入到中文信息检索中。但由于中西方语言的差异，不能按照英文的检索方法来检索中文文档。本文考虑了基于词表和基于二元两种不同的分词方法来切分中文文本信息，其中基于二元的方法采用的是重叠二元法，把 *ABCDEF* 切分为 *AB*、*BC*、*CD*、*DE* 和 *EF*。切分后的文档词与词之间用空格区分，在建立位置语言模型时用词来标识文档中的位置 i，而不再是一个单字一个位置，这样在考虑词与词之间位置近邻性的同时还在一定程度上考虑到了词间的复合关系，有助于提高检索的准确率。

文中分别基于词表索引和二元索引来完成中文文本检索任务，但由于文档中位置的数量远远大于文档的总数，因此如果只是简单地按上述所说的方法来实现位置语言模型，那么对位置语言模型的估计和排序的成本可能会非常高。因此，对于给定的查询，假设文档中的所有词项都拥有相同 σ 的传播函数，且该核密度函数的曲线是对称的，也就是说 $k(i,j)=k(j,i)$，那么就可以将整个建模过程中最耗时的部分即计算归一化的长度改写为：

$$\begin{aligned}\sum_{w\in V}c'(w,i)&=\sum_{w\in V}\sum_{j=1}^{N}c(w,j)k(i,j)\\&=\sum_{j=1}^{N}(\sum_{w\in V}c(w,j))k(i,j)\\&=\sum_{j=1}^{N}k(i,j)=\sum_{j=1}^{N}k(j,i)\end{aligned}$$

这意味着从位置 i 到位置 j 的传播数与从位置 j 到位置 i 的传播数相同。下面将展示如何利用高斯核函数来计算：

$$\begin{aligned}\sum_{j=1}^{N}k(j,i)&=\sum_{j=1}^{N}e^{\frac{-(j-i)^2}{2\sigma^2}}=\sqrt{2\pi}\sigma\sum_{j=1}^{N}\frac{1}{\sqrt{2\pi}\sigma}\cdot e^{\frac{-(j-i)^2}{2\sigma^2}}\\&\approx\sqrt{2\pi}\sigma\cdot\int_1^N\frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(j-i)^2}{2\sigma^2}}\,dj\\&=\sqrt{2\pi}\sigma\cdot[\Phi(\frac{N-i}{\sigma})-\Phi(\frac{1-i}{\sigma})]\end{aligned}$$

其中，$\Phi(\cdot)$ 是累积正态分布，N 表示文档长度。

从上面的分析可以看到，位置语言模型能够通过类似于固定长度的任意段落检索来实现。因此，可以使用文献[16]提出的段落检索技术来实现该模型，这样基于位置语言模型的文档排序就会与常规的文档排序拥有相同的复杂度。

3 实验及结果分析

3.1 实验数据集介绍及预处理

评测位置语言模型检索系统的性能所使用的实验数据集是 2 个常用的标准中文文档测试集 NTCIR5、NTCIR6。这类评测集合主要来源于 4 个港台主流报刊杂志的新闻语料，共 901446 篇文档。两个测试集分别提供了不同类型的 50 个中文查询主题，均由一个 XML 文档格式的文件提供，每个查询主题主要包含以下 5 个元素：主题编号 NUM、主题标题 TITLE、主题简单描述 DESC、主题详细描述 NARR、主题关键字 CONC。为了便于比较结果，一致采用主题标题作为查询条件，在查询时将查询结果中的前 1000 篇文档作为返回结果。

3.2 核函数与平滑方法的选取

本文考虑了 4 种有代表性的核函数：高斯、三角、余弦、和圆，以及两种常用的平滑模型：Dirichlet Prior 和 Jelinek-Mercer。在这里，将 Dirichlet Prior 平滑的参数设置为 1000，Jelinek-Mercer 平滑的参数设置为 0.5，而 4 个核函数所共有的参数变量 σ 设置为固定值 175。用每种平滑方法对比不同的核函数，通过对比结果的 MAP 值来选取更适合位置语言模型的平滑方法和核函数，对比结果如表 1 所列。由实验数据可以看出，不同核函数在 Dirichlet 先验平滑下的 MAP 值要普遍高于在 Jelinek-Mercer 平滑下所获得的 MAP 值；同时，

Dirichlet 先验平滑对 4 种核函数中高斯核函数的 MAP 值最高。因此，本文选取高斯核函数和 Dirichlet 先验平滑方法来构造位置语言模型。

表 1　不同核函数对应不同平滑方法的 MAP 值

Kernel Function	Dirichlet Prior	Jelinek-Mercer
Gaussian	0.2616	0.2488
Circle	0.2580	0.2525
Triangle	0.2561	0.2489
Cosine	0.2523	0.2102

3.3　两种索引方法在不同模型中检索效果的评估

通过信息检索评测可以评价不同技术的优劣以及不同因素对系统的影响。为了验证位置语言模型在中文信息检索中的有效性及不同索引方法对位置语言模型的影响，本文使用向量空间模型[17,18]、BM25 概率模型[19]、统计语言模型[20]和位置语言模型在语料库 NCTIR5 和 NTCIR6 上进行检索，采用 MAP、P@N、R-Precision 及 Recall 为评测指标。表 2 给出了 4 个检索模型在 2 个数据集上基于两种不同索引方法的检索平均准确率 MAP，其中百分比表示位置语言模型在 BM25 基础上增加的百分比。

表 2　4 个检索模型在 2 个数据集上基于两种索引方法的检索性能

Retrieval Models	NTCIR5		NTCIR6	
	word	bi-gram	word	bi-gram
TFIDF	0.2029	0.2299	0.1968	0.2023
BM25	0.2365	0.2508	0.2037	0.2123
KL	0.3079	0.3143	0.2189	0.2227
位置 lm	0.3232 (+36.66%)	0.3206 (+27.83%)	0.2444 (+19.98%)	0.2427 (+14.32%)

通过比较 4 个检索模型的 MAP 值可发现：位置语言模型在 2 个中文文档数据集上的检索准确率都要优于 TFIDF、BM25 和 KL，特别是在 NTCIR5 上提升较大。因此在本文使用的 2 个数据集上，位置语言模型都比传统的语言模型拥有更好的检索性能。

表 3、表 4 详细列出了在 NTCIR5 数据集下，不同检索模型在基于词表和基于二元两种索引方法下的不同评测指标的对比结果。

表 3　不同检索模型在基于词表索引方法下的评估结果

Evaluation	TFIDF	BM25	KL	位置 lm	%Δ
MAP	0.2029	0.2365	0.3079	0.3232	+36.66
P@10	0.3320	0.3600	0.4600	0.4980	+38.33
P@20	0.3000	0.3380	0.4110	0.4290	+26.92
R-Precision	0.2436	0.2655	0.3395	0.3395	+27.87
Recall	0.6730	0.6979	0.7169	0.7294	+4.51

表 4　不同检索模型在基于二元索引方法下的评估结果

Evaluation	TFIDF	BM25	KL	位置 lm	%Δ
MAP	0.2299	0.2508	0.3143	0.3206	+27.83
P@10	0.3580	0.3380	0.4520	0.4660	+37.87
P@20	0.3220	0.3330	0.4220	0.4300	+29.13
R-Precision	0.2604	0.2890	0.3317	0.3374	+16.75
Recall	0.7651	0.7533	0.7772	0.7824	+3.86

表 3、表 4 中列%Δ的数据为位置语言模型比 BM25 概率模型性能提高的百分比值。与 BM25 概率模型相比，在两种不同索引方法下位置语言模型模型的平均准确率分别提高了 36.66%和 27.83%，召回率分别提高了 4.51%和 3.86%。从上表的对比实验结果可以看出，位置语言模型明显改善了检索系统的性能。两种不同索引方法下 4 组模型结果的对比关系可以从图 2、图 3 中更直观地看到。

从实验结果可以看出，基于位置语言模型的检索系统由于将词项近邻性和段落检索结合在一起，更加细致地刻画了文档中词与词之间的位置关系，因此能够比较明显地提高检索系统的性能。当然另外一方面，由于要对文档中的每个位置单独建立语言模型，因此运算时间等开销会很大。但是在实际实现过程中，为了提高效率，只计算出现了查询词项的位置的得分，因为直觉上我们认为最佳匹配位置应该是靠近这些位置的，从而在不对检索结果产生太大影响的前提下还大大降低了运算的时间复杂度。

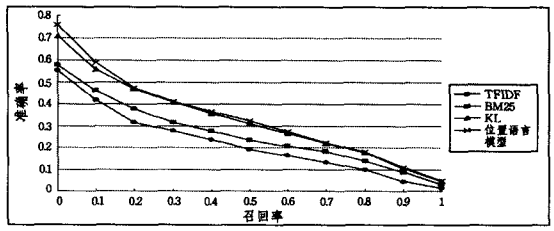

图 2　基于词表索引方法下 4 个检索模型的召回率-准确率

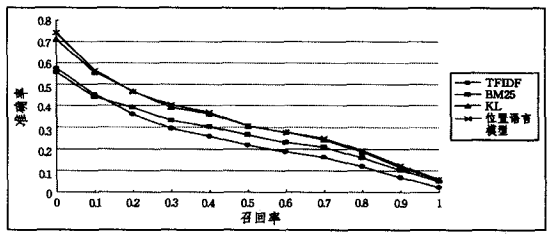

图 3　基于二元索引方法下 4 个检索模型的召回率-准确率

3.4　两种索引方法在位置语言模型中的对比结果

前面的实验分别比较了在两个数据集下基于两种不同索引方式的 4 种检索模型的检索效果，发现不论是在词表索引下还是在二元索引下，位置语言模型的检索性能都要明显优于其他 3 种传统的检索模型。而对于位置语言模型，两种索引方法又各自有优缺点，表 5 列出了在 rigid 和 relax 测试条件下两种索引方法对位置语言模型性能的影响。

表 5　不同索引方法对位置语言模型的影响

Evaluation	rigid		relax	
	bi-gram	word	bi-gram	word
MAP	0.2547	0.2616	0.3206	0.3232
P@10	0.3240	0.3740	0.4660	0.4980
P@20	0.2920	0.3020	0.4300	0.4290
R-Precision	0.2732	0.2879	0.3374	0.3395
Recall	0.7814	0.7279	0.7824	0.7294

从表 5 中可以看出，在 rigid 和 relax 两种测试条件下词表索引的准确率普遍要高于二元索引，但是二元索引的召回率又要高于词表索引。其中在 rigid 和 relax 集合下，词表索引相对二元索引的 MAP 值分别提高了 2.71%和 0.81%，而二元索引的召回率分别提高了 7.35%和 7.27%，由此可以看出两种索引方式对位置语言模型的检索性能没有太大的影响。不过从 P@10 精度的对比可以看出，基于词表索引下的检索结果中排名靠前的文档中相关文档的数量得到了明显的增加，在两种测试条件下 P@10 分别提高了 15.43%和 6.87%，使得相关文档能够更加集中地出现在靠前的位置。

上面的实验比较了两种索引方法对位置语言模型性能的整体影响，发现词表索引拥有较好的准确率，二元索引则拥有更高的召回率，但对于每个具体的查询主题，不同的索引方法又有不同的影响。图 4 显示了针对 NTCIR5 数据集，两种不

同索引方法下50个查询主题的准确率对比情况。从图4中可以看到，针对每一个具体的查询，两种索引方法的检索效果存在一定的差异，究其原因，我们认为这个差异是由分词准确性引起的。例如查询[018]"菸草商，诉讼赔偿"，二元索引的检索准确率为0.4757，要高于词表索引的准确率0.1749，这是因为词"菸草"并没有存储在词表中，当用词表方法分词时，这个词就会被分开，而用二元分词时它就能被识别为一个词，这样就能大大提高检索精度。但是在现实文档中，很多词都是由两个或多个简单的词组成的，把这样的长词作为一个词处理，效果可能会更好。对于查询[017]"印度与巴基斯坦领土冲突"，词表索引就拥有更好的检索效果。因为词"巴基斯坦"存储在词表中，分词时会把它作为一个词处理，但是二元分词就会将其细分为多个简单词，这样就会降低检索的精度。因此如何处理复杂词和简单词之间的矛盾冲突仍然是个开放性的问题，需根据具体的模型需求选择更适合的索引方法。

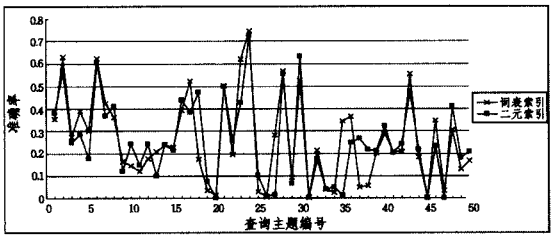

图4　两种索引方法下50个查询主题的准确率

结束语　鉴于位置语言模型在Trec语料检索中取得了较好的检索效果，将其引入到中文信息检索中，通过对NTCIR5、NTCIR6语料采用基于词表和基于二元的两种中文文本切分方法，使其适合于位置语言模型的建模特点。对于文档中的位置定义，不是以单个的字为单位，而是分词后的一个词项标识一个位置，从而充分利用了中文文本的特色，使其获得更好的检索效果。文中比较了基于词表和基于二元两种中文索引方法分别在位置语言模型和传统检索模型中的检索效果，另外还重点对比了两种索引方法在位置语言模型中的应用差异。实验结果证明，位置语言模型在中文信息检索上也能取得不错的效果，并且两种中文索引方法在位置语言模型上的应用也都能得到较好的检索性能。

参 考 文 献

[1] Ponte J, Croft W B. A Language Modeling Approach to Information Retrieval[C]// Proceedings of the 1998 ACM SIGIR Conference on Research and Development in Information Retrieval. Melbourne, 1998: 275-281

[2] Lv Yuan-hua, Zhai Cheng-xiang. A comparative study of methods for estimating query language models with pseudo feedback[C]// Proceedings of 2009 CIKM Conference on Information and Knowledge Management. HongKong, 2009: 1895-1898

[3] Diaz F, Metzler D. Improving the estimation of relevance models using large external corpora[C]// Proceedings of the 2006 ACM SIGIR Conference on Research and Development in Information Retrieval. Washington, 2006: 154-161

[4] Liu Xiao-yong, Croft W B. Cluster-based retrieval using language models[C]// Proceedings of the 2004 ACM SIGIR Conference on Research and Development in Information Retrieval. Sheffield, 2004: 186-193

[5] Lv Yuan-hua, Zhai Cheng-xiang. Positional language models for information retrieval[C]// Proceedings of the 2009 ACM SIGIR Conference on Research and Development in Information Retrieval. Boston, 2009: 299-306

[6] 余伟，王明文，万剑怡，等. 结合语义的位置语言模型[J]. 北京大学学报(自然科学版)，2013，49(2)：203-212
Yu Wei, Wang Ming-wen, Wan Jian-yi, et al. Positional language models with semantic information[J]. Journal of Peking University(Natural Science Edition), 2013, 49(2): 203-212

[7] Miao Jun, Huang Xiang-ji, Ye Zheng. Proximity-based rocchio's model for pseudo relevance[C]// Proceedings of the 2012 ACM SIGIR Conference on Research and Development in Information Retrieval. Portland, 2012: 535-544

[8] Lv Yuan-hua, Zhai Cheng-xiang. Positional relevance model for pseudo-relevance feedback[C]// Proceedings of the 2010 ACM SIGIR Conference on Research and Development in Information Retrieval. Geneva, 2010: 579-586

[9] Kwok K L. Comparing representations in Chinese information retrieval[C]// Proceedings of the 1997 ACM SIGIR Conference on Research and Development in Information Retrieval. 1997: 34-41

[10] Lam W, Wong C Y, Wong K F. Performance evaluation of character, word and n-gram-based indexing for Chinese text retrieval [C]// Proceedings of the Information Retrieval with Asian Languages 97 Conference. 1997: 68-80

[11] Nie J Y, Ren F. Chinese information retrieval: using characters or words[J]. Information Processing and Management, 1997, 35(4): 443-462

[12] Zhai Cheng-xiang, Lafferty J D. A study of smoothing methods for language models applied to ad hoc information retrieval[C]// Proceedings of the 2001 ACM SIGIR Conference on Research and Development in Information Retrieval. New Orleans, 2001: 334-342

[13] Zhao Jia-shu, Huang Xiang-ji, He Ben. CRTER: using cross terms to enhance probabilistic information retrieval[C]// Proceedings of the 2011 ACM SIGIR Conference on Research and Development in Information Retrieval. Beijing, 2011: 155-164

[14] Kise K, Junker M, Dengel A, et al. Passage Retrieval Based on Density Distributions of Terms and Its Applications to Document Retrieval and Question Answering [M]. Reading and Learning: Adaptive Content Recognition. 2004: 306-327

[15] Petkova D, Croft W B. Proximity-based document representation for named entity retrieval[C]// Proceedings of the 2007 CIKM Conference on Information and Knowledge Management. Lisboa, 2007: 731-740

[16] Kaszkiel M, Zobel J, Sacks-Davis R. Efficient passage ranking for document databases[J]. ACM Transactions on Information Systems, 1999, 17(4): 406-439

[17] Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-620

[18] Salton G, Fox E A, Wu H. Extended Boolean information retrieval[J]. Communications of the ACM, 1983, 26(11): 1022-1036

[19] Maron M E, Kuhns J L. On relevance, probabilistic indexing and information retrieval[J]. Journal of the ACM(JACM), 1960, 7(3): 216-244

[20] Berger A, Lafferty J. Information retrieval as statistical translation[C]// Proceedings of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval. Berkley, 1999: 222-229