基于特征选择和语义扩展的词序列核函数研究

刘克彬[1]，李芳[2]，刘磊[3]，韩颖[4]

（1, 2, 3, 4. 上海交通大学计算机科学与工程系，上海市 200240）

摘 要： 词序列核函数是 Convolution 核的一种，它处理的对象是离散的词序列。词序列核函数的计算过程不需要显式地构造特征向量，而且具有良好的复合特性，在自然语言处理领域有广泛的应用。传统的词序列核函数没有考虑到语义信息，本文对传统的词序列核函数加入了特征选择和语义扩展，得到一种新的核函数。本文使用该新函数进行实体关系自动抽取的实验，并与基于特征向量的方法和传统的词序列核函数作了比较。结果证明新函数的效果好于传统的方法，特别是在小规模训练集合上面具有较大的优势。

关键词： 核方法；语义；特征选择；词序列核函数

Research on Word-Sequence Kernel with Feature Selection and Semantic Extension

Kebin Liu[1], Fang Li[2], Lei Liu[3], Ying Han[4]

(1,2,3,4. Dept. of Computer Science & Engineering, Shanghai Jiaotong University, Shanghai 200240)

Abstract: Word-Sequence Kernel is one kind of the Convolution kernels; it works on the discrete word sequences. Word-Sequence kernel has many fine properties such as the combination property and has been widely used in natural language processing field. But traditional Word-Sequence kernel failed to consider the semantic information. This paper proposed a new kernel function which embeds the semantic information and feature selection into kernel function calculation. The experiments of relation extraction were carried out to compare the new kernel function with the feature-based approach and traditional Word-Sequence kernel. The conclusion is that, the new kernel function has better generalization ability and gains a better performance than other two traditional methods especially on a small training set.

Key words: Kernel Method; Semantic; Feature selection; Word-Sequence Kernel

1 引言

近年来基于核的方法越来越受到人们的重视，它最初是在支持向量机(SVM)中被引入，现在已有多种基于核的学习算法并成功应用于分类、回归等应用。其基本原理是使用核函数来代替传统的学习算法中的向量内积运算。典型的核函数具有以下的形式：假设 X，Y 是原空间中的两个向量，核函数的目的在于寻找一个映射 Φ，将 X，Y 映射到 Φ(X)，Φ(Y)，核函数的值就是新的特征空间中两个映射向量的内积，K(X, Y) =Φ(X)•Φ(Y)。

目前常用的核函数大致可以分为两大类，一类以引入先验知识为主要目的，例如潜在语义(LSI)核函数[1]，主成分分析(PCA)核函数[2]等。这些方法没有考虑词的先后顺序，因此无法捕捉到一些顺序带来的信息。运行在离

基金资助：上海市科委国际合作项目“基于 INTERNET 信息的智能化检索”(045107035)，该研究同时得到德方的资助。

作者简介：刘克彬(1981-)，男，山东省青岛市，硕士在读，captainlkb2003@sjtu.edu.cn

散的数据结构上的 convolution 核解决了这个问题，词序列核函数是其中重要的一种。在词序列核函数计算的过程中，保持原来的词序列不变，即不需要显式的构造特征向量，而是在两个输入词序列之间寻找公共子序列，这些子序列都是有序的，有效利用了顺序和结构信息。因为目标映射空间的维度是由全体子序列来索引的，所以潜在的映射后的空间维数可能是无限的。但是词序列核函数仅计算其中有限维，所以这种核函数具有良好的计算效率和实用性。目前的词序列核函数还存在缺乏语义支持的不足，在小训练集上面的泛化能力不够。为了解决上述问题，本文提出了一种改进的加入语义信息和特征选择的词序列核函数，在小训练集的情况下能够取得比较好的效果。这种核函数基于传统的词序列核函数[3]并进行了改进。

本文的其余部分介绍的内容如下：第二部分介绍了词序列核函数的一些相关工作。第三部分介绍了词序列核函数的原始实现和改进方法。第四部分是实验和结果的分析。文章最后一部分作了总结和对下一步工作的展望。

2 相关工作

词序列核函数是 convolution 核的一种，Haussler [4]于 1999 年提出了在离散数据结构上建立核函数的方法，揭开了研究 convolution 核的序幕。与树结构核函数[5,6]和图结构核函数[7]不同，词序列核函数以词序列作为操作的对象。Lodhi[8]提出一种序列核函数以及一种模拟的实现方法来进行文本分类工作，并与 NGK(N 元组核)等传统方法作了比较，所用序列基本单位是字符。cancedda [3]引入词序列的核函数，计算过程中寻找两个词序列中的公共子序列，子序列中允许间隔存在。他同时提出了一些改进，包括使用多种不同的衰减因子以及用 GVSM(General Vector Space Model)来进行软匹配以及多语种处理的方向。Suzuki[9]对词序列核函数作了改进，使用 χ^2 统计量来进行特征选择。本文的研究对词序列核函数[3,8]做了两方面的扩展，一方面在核函数计算过程中嵌入语义知识，本文改进了语义知识的获取方法和语义知识嵌入的实现方法。另一方面是使用信息熵来进行特征选择。语义知识来源于本体，特征选择的依据是信息熵理论。语义嵌入和特征选择提高了核函数的泛化能力，取得了很好的效果。

3 词序列核函数

3.1 传统词序列函数的实现

定义 Σ 为中文词汇的集合，在此集合上定义词的序列 $S = S_1S_2\cdots S_{|S|}$。 $\mathbf{i} = [i_1, i_2,\cdots,i_n]$表示 S 的索引的一个子集，其中 $1 \leqq i_1<i_2<\cdots<i_n \leqq |S|$，则 $S[\mathbf{i}] \in \Sigma^n$ 是 S 的子序列。$l(\mathbf{i})$表示 S[**i**]在原序列中跨过的宽度，包含间隔。n 是 S[**i**]的词数。例如：假设"ACDBABC"为一个词序列，每个大写字母代表一个词，n=3 时，假设要寻找包含"ADB"的子序列，原序列中的"ACDB"和"ACDBAB"都将入选。他们的索引序列分别为[1,3,4]和[1,3,6]，在原序列中跨过的宽度分别为 3 和 5。

词序列核函数的基本思想是寻找并统计两个词序列中的公共子序列数量作为核函数的结果，考虑到子序列中可能包含间隔项，对找到的公共子序列使用衰减因子来计算权重(基于如下假设：包含间隔越多的子序列对整体相似度的贡献越小)，基本实现公式如下：

$$K_n(S,T) = \sum_{u\in\Sigma^n}\sum_{\mathbf{i}:u=S[\mathbf{i}]}\sum_{\mathbf{j}:u=T[\mathbf{j}]}\lambda^{l(\mathbf{i})+l(\mathbf{j})} \tag{1}$$

其中 u 是公共的子序列，通过三层循环统计所有这样的公共子序列。S[**i**]和 T[**j**]都可能是不连续的，因为衰减因子 λ 的存在，子序列跨越的距离越大权重就越小。以下是递归实现的过程：

$$K_n(Sa,T) = K_n(S,T) + \sum_{j:T_j=a}\lambda^2 K'_{n-1}(S,T[1:j-1]) \tag{2}$$

$$K'_i(Sa,T) = \lambda K'_i(S,T) + K''_i(Sa,T) \tag{3}$$

$$K''_i(Sa,Tb) = \lambda K''_i(Sa,T) + \lambda^2 K'_{i-1}(S,T)\delta(a,b) \tag{4}$$

$$K_n(S,T)=0, if \min(|S|,|T|)<n \tag{5}$$

$$K'_i(S,T)=0, if \min(|S|,|T|)<i,(i=1,...,n-1) \tag{6}$$

$$K''_i(S,T)=0, if \min(|S|,|T|)<i,(i=1,...,n-1) \tag{7}$$

$$K'_0(S,T)=1 \tag{8}$$

3.2 改进的基于特征选择和语义扩展的词序列函数实现

3.2.1. 词序列核函数的语义信息嵌入

本文使用基于本体的语义知识[10]来衡量词语之间的语义相似度，采用的本体是 hownet[11]。本文没有用词序列作为核函数的输入，而是充分利用词性标注的结果，将核函数的输入变为词性加词条的双序列结构。寻找公共的词性子序列的计算过程中嵌入对应的词条子序列的语义相似度。对于匹配项和间隔项使用不同的衰减因子 λ_m 和 λ_g。新的核函数需要一些额外定义，首先定义双序列 $X = X_1X_2...X_{|X|}$，每个词定义一个二元组 $X_i=(p, w)$，p 代表了词 X_i 的词性，w 表示 X_i 的词条。$\mathbf{i} = [i_1, i_2, ... i_n]$和 $l(\mathbf{i})$的定义同上。语义信息嵌入相当于作如下的映射：

$$K_n(X,Y)=\phi(X.p)^T S(Y.p) \tag{9}$$

其中Φ是映射函数，S 语义词序列语义相似度矩阵。映射和矩阵都不需要显式的构建，该过程均在核函数的计算过程中来完成。核函数公式修改如下：

$$K_n(X,Y)=\sum_{u\in\sum^n i:u=X[\mathbf{i}].p}\sum_{\mathbf{j}:u=Y[\mathbf{j}].p}\lambda_m^{2n}\prod_{k=1}^{n}SIM(X_{i_k}.w,Y_{j_k}.w)\prod_{i_1<l<i_n,l\in\mathbf{i}}\lambda_g\prod_{j_1<h<j_n,h\in\mathbf{j}}\lambda_g \tag{10}$$

SIM 函数用来计算语义相似度，两种不同的衰减因子分别代表匹配项和间隔项的权重。以下是新的核函数的递归计算公式：

$$K_n(Xa,Y)=K_n(X,Y)+\sum_{j:Y_j.p=a.p}\lambda_m^{\ 2}K'_{n-1}(X,Y[1:j-1])SIM(a.w,Y_j.w) \tag{11}$$

这是对公式(2)改进后的 K_n 公式，a 是确定出现在所有的公共子序列里面的匹配项，因此使用 λ_m 作为它的权重。

$$K'_i(Xa,T)=\lambda_g K'_i(X,Y)+K''_i(Xa,Y) \tag{12}$$

$$K''_i(Xa,Yb)=\lambda_g K''_i(Xa,Y)+\lambda_m^{\ 2}K'_{i-1}(X,Y)SIM(a.w,b.w)\delta(a.p,b.p) \tag{13}$$

$$K_n(X,Y)=0, if \min(|X|,|Y|)<n \tag{14}$$

$$K'_i(X,Y)=0, if \min(|X|,|Y|)<i,(i=1,...,n-1) \tag{15}$$

$$K''_i(X,Y)=0, if \min(|X|,|Y|)<i,(i=1,...,n-1) \tag{16}$$

$$K'_0(X,Y)=1 \tag{17}$$

3.2.2. 特征选择和结果嵌入

通过特征选择，区别对待对分类贡献不同的特征。本文使用信息熵衡量特征。将特征在各类别中出现与否看

作一个随机事件，频繁出现在某单独类别中的特征具有强的代表性。信息熵的计算公式如下：

$$Entropy(T) = \sum_{i=1}^{N} - p_i \log_2 p_i + \gamma \tag{18}$$

N 代表了类别的总数，p_i 代表了特征在每个类别出现的概率，γ 是常数，因为信息熵可能出现零值，因此设置这个参数对结果进行平滑处理。p_i 通过特征项在训练集合中的出现频度来计算。对于训练集合中未出现的词，将其信息熵赋值为一个较大的常数。对上一节中的核函数公式(11)，(13)进行修改以嵌入特征选择的结果：

$$K_n(Xa, y) = K_n(X, Y) + \sum_{j:Y_j.p=a.p} \frac{\lambda_m{}^2 K'_{n-1}(X, Y[1:j-1]) SIM(a.w, Y_j.w)}{Entropy(a) Entropy(Y_j)} \tag{19}$$

$$K''_i(Xa, Yb) = \lambda_g K''_i(Xa, Y) + \frac{\lambda_m{}^2 K'_{i-1}(X, Y) SIM(a.w, b.w) \delta(a.p, b.p)}{Entropy(a) Entropy(b)} \tag{20}$$

修改之后的核函数的计算复杂度依然是 O(n|X||Y|)，其中 X，Y 分别是输入序列。

4 方法评测

实验使用从 Web 上收集的文档，系统从中自动生成出 2819 个候选关系，人工标注后作为训练和测试集合。关系抽取实验对以下三种方法作了比较。方法 1 是传统的基于特征向量的方法，通过向量的内积来计算对象之间的相似度。向量每一维的权重采用 TF-IDF 来计算。方法 2 是传统的词序列核。方法 3 是改进后的语义词序列核。设计两组实验，分别重复多次并对结果进行了平均处理。

4.1 实验结果及分析

为了考察改进后的核函数性能，实验一对几种方法作了比较。表 1 是实验结果：

表 1 关系抽取结果比较

Tab. 1 Results of Relation Extraction

	正确	错误	正例总和	正确率	召回率	F 测度
基于特征向量（方法 1）	351	57	477	86.0%	73.6%	79.3%
传统词序列核（方法 2）	366	56	477	86.7%	76.7%	81.4%
语义词序列核（方法 3）	359	39	477	90.2%	75.3%	82.1%

从结果可以看出新的核函数有一定优势，这是在使用大训练集得到的结果。为了验证新的核函数的泛化能力，进行第二组实验：训练集合的规模每次递减，随机抽取 100%，80%，60%，40%，20%，10%的训练样例来进行训练，记录各种算法的表现：

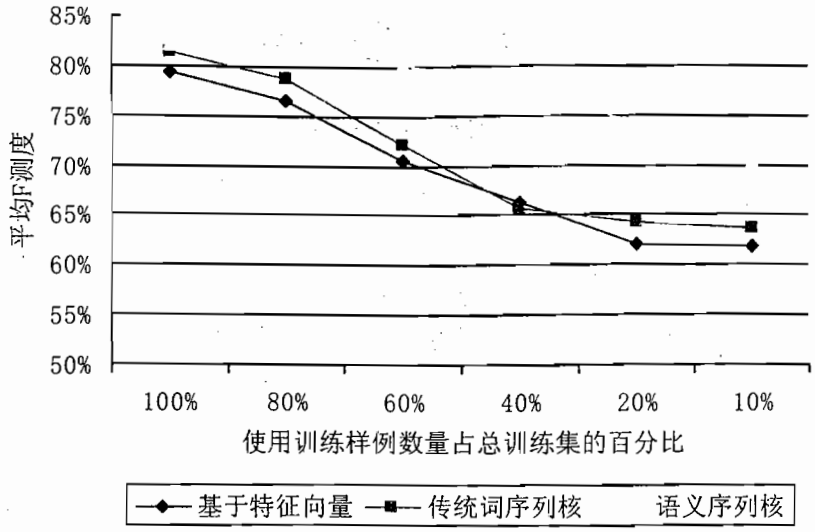

图 1 不同规模训练集下的关系提取结果

Fig.1 Relation extraction result with various size of training sets

由以上结果可以发现新的核函数的确有更好的泛化能力。即使是在只有 20%训练语料的情况下依然有比较好的效果。其他两种方法在训练集合规模减小到 50%的时候准确率和召回率有明显的下降。

5 总结和展望

本文提出了一种在词序列核函数嵌入语义知识和特征提取结果的方法，得到一种新的核函数。采用词性和词条的双序列结构，改进了传统词序列核函数。语义知识的计算基于中文的本体 hownet，特征选择采用了信息熵作为度量标准。改进的方法具有比较好的效果和泛化能力，尤其是在训练集比较小的情况下有很好的效果。但是语义的计算和特征提取都有计算的开销，设计更加有效的语义获取算法是未来要努力的方向。另外，下一步的工作还可以考虑通过核函数的复合引入更多的有效信息。

参考文献：

[1] Cristianini N, Shawe-Taylor, J, Lodhi, H. Latent Semantic Kernels[J]. Journal of Intelligent Information Systems, 2002, 18:2-3.

[2] Schölkopf B, Smola A, Müller K-R. Kernel Principal Component Analysis[J]. MIT Press, 1999, 327–352.

[3] Cancedda N, Gaussier E, Goutte C, et al. Word-Sequence Kernels[J]. Journal of Machine Learning Research, 2003, 3:1059–1082.

[4] Haussler D. Convolution kernels on discrete structures[R]. Technical Report UCSC-CRL-99-10, 1999.

[5] Zelenko D, Aone C, Richardella A. Kernel Methods for Relation Extraction[J]. Jour Journal of Machine Learning Research, 2003, 1083-1106.

[6] Collins M, Duffy N. Convolution Kernels for Natural Language[C]. Proc. NIPS-2001, 2001.

[7] Suzuki J, Hirao T, Sasaki Y, et al. Hierarchical Directed Acyclic Graph Kernel: Methods for Structured Natural Language Data[C]. Proc. of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-2003), 2003.

[8] Lodhi H, Saunders C, Shawe-Taylor J, et al. Text Classification using String Kernels[J]. Journal of Machine Learning Research, 2002, 2:419–444.

[9] Suzuki J, Isozaki H, Maeda E. Convolution Kernels with Feature Selection for Natural Language Processing Tasks[C]. Proc. 42nd Meeting of Association for Computational Linguistics, 2004.

[10] 刘群，李素建. 基于《知网》的词汇语义相似度计算[A]. 第三届汉语词汇语义学研讨会，2002.

[11] 董振东，董强. 关于知网-中文信息结构库. http://www.keenage.com