文章编号: 1001-0920 (2016) 09-1711-04 DOI: 10.13195/j.kzyjc.2015.0793

基于张量的 XML 相似度计算方法

朴 勇, 江 贺, 王秀坤

(大连理工大学 软件学院，辽宁 大连 116620)

摘 要: 扩展标记语言 (XML) 带有一定的结构和语义信息, 与普通文本相比, XML 具有描述精确、表现形式丰富等特点, 但同时也使得传统的自然语言处理和数据挖掘等技术不能直接应用. 根据 XML 内容和结构并非独立, 内容影响结构, 结构作用于内容, 提出一种基于张量的 XML 特征降维及综合相似度计算方法. 针对 XML 文档, 使用张量表示并采用基于最大互信息的方法对其进行降维, 采用将 XML 结构和内容相融合的综合相似度度量方法确定结构和内容的内在联系及共同作用方式, 提高 XML 综合相似度计算性能. 实验及结果分析验证了所提出方法的有效性.

关键词: 扩展标记语言；综合相似度；张量分析；特征降维

中图分类号: TP311 **文献标志码:** A

Tensor-based approach to XML similarity calculation

PIAO Yong, *JIANG He*, *WANG Xiu-kun*

(School of Software，Dalian University of Technology，Dalian 116620，China．Correspondent：PIAO Yong，E-mail: piaoy@dlut.edu.cn)

Abstract: XML documents have both structural and semantic information, bringing data integration and deeply utilization based on XML more precise description and versatile expression, but meanwhile traditional natural language processing(NLP) and data mining(DM) methods can not be applied directly. Feature dimension reduction and general similarity of XML based on tensor analysis are discussed. Considering the correlation between XML's structure and content, a tensor based method of describing XML documents and a maximization mutual information(MMI) method of XML's dimension reduction are presented. Since the structure and the content are not independent each other, a tensor based algorithm of calculating general similarity from a non-linear angle is designed to show their relationships and effects, which can improve the calculated performance for the general similarity of XML. The experimental results show the effectiveness of the proposed method.

Keywords: XML；general similarity；tensor analysis；feature reduction

0 引 言

在互联网高速发展的今天, 以 XML 为代表的半结构化技术已成为主要的数据交换和传输格式, 是很多文档分类及聚类分析的主要研究对象. XML 因自身结构的特点, 为文档的分析和挖掘提供了另一个层面的线索. 识别 XML 文档间的相似度或包含性等工作在很大程度上不同于传统的基于文本的分析方法, 使得传统的方法不能直接应用于 XML 的文档处理.

由于 XML 文档中嵌入了结构信息, 内容和结构对 XML 的处理都有着重要作用[1]. XML 的文本内容和结构信息是 XML 文档不可或缺的特征, 是进行 XML 完整分析和利用的基础[2-4], 因此, 在计算 XML 文档相似度信息时需要将两者结合起来综合考虑, 即综合相似度. XML 的综合相似度计算普遍采用两种方式: 线性综合法和非线性综合法. 线性综合法分别计算内容和结构相似度, 然后通过权重参数对结构和内容相似度进行线性混合; 而非线性综合法则将内容和结构相似度体现在同一模型中处理, 忠于 XML 原始信息的构成.

Guo 等[5]采用线性综合法, 通过不同向量空间模型分别对内容特征和结构特征进行表示, 结构特征为全体路径的集合, 通过计算两向量的乘积获得距离. 文献 [6] 也是分别计算路径相似度和内容相似度, 然后对结构和内容赋予一定的权值, 最后计算总和作为最终相似度.

结构链接向量模型 (SLVM)[7] 是一种采用非线性

收稿日期: 2015-06-23；**修回日期:** 2015-09-22.

基金项目: 国家自然科学基金项目(61370144).

作者简介: 朴勇(1975–), 男, 副教授, 博士, 从事数据挖掘与智能计算等研究；江贺(1980–), 男, 教授, 博士生导师, 从事软件工程、数据挖掘等研究.

综合法来计算文档整体相似度的模型, SLVM 通过不同的向量空间模型表示文档的结构和内容特征, 然后使用向量链接将两者对应. 对不同文档集的实验结果表明, 该模型的F测度可以稳定在0.82～0.85. SLVM的一个主要的特点是考虑了文档集中文档间的结构引用关系, 但是, 当文档较为独立时, 该模型将退化为主要考虑文档的内容相似度.

Yoon 等[8]给出了另外一种非线性模型, 将文档、路径和内容项的关系通过一种3维矩阵进行描述, 在进行文档间相似度的计算时, 该模型使用了“位异或”的操作方式, 并且考虑了结构路径的popularity特征. 该特征可以理解为结构表示的路径频率, 但路径节点的位置信息在模型中并未考虑.

综上所述, 非线性综合法将结构和内容特征在同一模型中表示, 因此在一定程度上具有捕获结构和内容间相关性的能力, 在结果的精度上也具有一定优势.

1 XML 的张量表示与特征降维

在结构相似度层面上, 由于基于路径的相似度计算方法结构简单, 算法性能也较高, 在处理大数据的应用中受到了更多的关注, 但如何提高路径匹配的精度是一个首要问题.

在综合相似度层面上, 传统的向量空间模型不能适应XML特有的结构特征和内容特征, 使得在此基础上的各种算法不能直接应用; 另外一个问题是这种方法会造成特征向量的维度过高, 使得计算复杂性过大, 如果特征项中还有结构路径信息, 则高维度的问题将更加严重.

采用基于张量的非线性综合分析方法(SCTA)对XML结构和内容进行综合考虑, 不仅使得计算准确高效, 而且能避免线性综合法中额外的经验参数给结果带来的直接影响.

1.1 张量表示

对于多维数据的混合处理, 采用张量表示是一种自然方法, 每个XML文档可表示为一个3阶张量 $\boldsymbol{\mathcal{X}} \in \boldsymbol{R}^{I_1 \times I_2 \times I_3}$ 的形式, 其元素 x_{i_1,i_2,i_3} 为对应条目在文档中出现的次数, 该条目由下标 i_1, i_2, i_3 决定, 分别代表3个维度上的坐标, 即所在的结构路径、路径节点和内容单词. 例如/books/book/title/“computer”, 其3个维度的坐标可分别为: 所在结构路径, 其取值范围为文档集合中包含的所有完整路径, 这里可为/books/book/title/subtitle; 路径节点, 其取值范围为所有DTD中定义的标签集合, 这里为title; 单词“computer”, 其取值范围为所有文本单词的集合.

在构建文档张量表示的时候: 如果某节点未在路径中出现, 则对应值标记为0值; 如果该节点下没有任何文本内容, 则以单词“NULL”表示并统计其元素值. 为了保留文档的层次结构, 每个节点的祖先节点需要依次重复记录该节点中的信息. 另外, 如果两个路径 P_1 和 P_2 具有相同的根节点, 且 P_1 是 P_2 的子序列, 则将 P_1 加入到路径集合中, 而忽略 P_2. 在所有符合 P_2 的路径中, 其他节点内容前置于 P_1 的叶子节点上, 与其原有内容一起进行记录, 这样做的主要目的是降低计算上的复杂程度, 从节点位置的角度考虑, 越靠后的节点对结果的影响越小. 这样的表示方法另外一个好处是能够将路径中包含的所有节点的内容一同记录下来, 包括中间节点. 由于每个路径中离根节点越近的节点出现的频率也相对要高, 张量模型中同时考虑节点位置和内容的因素.

1.2 特征降维

由于受到多种因素的影响, 张量数据一般具有更高的维度, 使得直接计算相似度非常困难, 而且本文研究的XML文档对象并不同构, 路径、节点及词条数目会很大, 即 $\boldsymbol{\mathcal{X}}$ 位于高维张量空间并且稀疏, 这将直接影响算法的效率和结果. 对此, 首先要对张量文档进行降维.

与传统的线性降维方法一样, 张量数据的降维也有无监督和有监督两类方法, 并且张量在降维过程中不会破坏数据原始的结构, 比传统方法提高了计算的效率和结果的精度[9]. 张量降维的思想是在尽量不丢失信息的情况下, 将高维张量映射到低维张量空间, 即将高维张量 $\boldsymbol{\mathcal{R}} \in \boldsymbol{R}^{I_1 \times I_2 \times \cdots \times I_n}$ 表示为 $\boldsymbol{\mathcal{R}} \approx \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \cdots \times_n \boldsymbol{U}^{(n)}$. 其中: $\boldsymbol{\mathcal{C}} \in \boldsymbol{R}^{R_1 \times R_2 \times \cdots \times R_n}$ 为维度经过筛选后的核心张量, 可以代表各维度之间的相关度, 同时有 $R_i \ll I_i$; $\times_n$ 为张量与矩阵第 n 模的乘积; $\boldsymbol{U}^{(n)}$ 为在 n 模空间上的转换矩阵, 一般为正交形式. $\boldsymbol{\mathcal{C}}$ 和 $\boldsymbol{U}^{(n)}$ 的求解一般不能通过解析形式得到, 高效的求解方法仍是目前的研究热点.

本文采用基于最大化互信息(MMI)的方法实现张量文档的降维, MMI是一种有监督的特征抽取方法. 令随机变量 $y \in \{1, 2, \cdots, C\}$ 为对应的类别, 通过Tucker分解对 $\boldsymbol{\mathcal{X}}$ 采用正交矩阵和核心张量表示, 即

$$\boldsymbol{\mathcal{F}} = \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{U}^{(1)\mathrm{T}} \times_2 \boldsymbol{U}^{(2)\mathrm{T}} \times_3 \boldsymbol{U}^{(3)\mathrm{T}} \in \boldsymbol{R}^{R_1 \times R_2 \times R_3}. \tag{1}$$

核心张量 $\boldsymbol{\mathcal{F}}$ 是 $\boldsymbol{\mathcal{X}}$ 的特征张量, 可以代替 $\boldsymbol{\mathcal{X}}$ 参与后续的分类或聚类的计算. 对于所有转换矩阵 $\{\boldsymbol{U}\}$ 同时进行求解比较困难, 可参照张量分解算法ALS的原理, 每次只针对其中一个求解. 假定求解 $\boldsymbol{U}^{(1)}$, 令 $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_2 \boldsymbol{U}^{(2)\mathrm{T}} \times_3 \boldsymbol{U}^{(3)\mathrm{T}}$, 则有 $\boldsymbol{\mathcal{F}} = \boldsymbol{\mathcal{Z}} \times_1 \boldsymbol{U}^{(1)\mathrm{T}}$, 对应的矩阵表示为 $\boldsymbol{F}_{(1)} = \boldsymbol{U}^{(1)\mathrm{T}} \boldsymbol{Z}_{(1)}$.

采用MMI的优化方式得到求解的目标方程, 即

$$\boldsymbol{U}^{(1)*} = \underset{\boldsymbol{U}^{(1)\mathrm{T}}\boldsymbol{U}^{(1)}=\boldsymbol{I}}{\arg\max}\ I(\boldsymbol{F}_{(1)}, y) = \underset{\boldsymbol{U}^{(1)\mathrm{T}}\boldsymbol{U}^{(1)}=\boldsymbol{I}}{\arg\max}\ I(\boldsymbol{U}^{(1)\mathrm{T}}\boldsymbol{Z}_{(1)}, y). \tag{2}$$

其中: $\boldsymbol{U}^{(1)\mathrm{T}} \in \boldsymbol{R}^{R_1 \times I_1}$, $\boldsymbol{Z}_{(1)} \in \boldsymbol{R}^{I_1 \times R_2 R_3}$. 由于 $\boldsymbol{F}_{(1)}$

为随机矩阵, 不能直接计算其 I 值, 可以参照采用其近似值 $\tilde{I}$ 的方法来简化计算, 即

$$\tilde{I} = \tilde{I}(\boldsymbol{U}^{(1)\mathrm{T}}\boldsymbol{Z}_{(1)}, y) = \sum_{r_1=1}^{R_1} \tilde{I}(\boldsymbol{u}_{r_1}^{(1)\mathrm{T}}\boldsymbol{Z}_{(1)}, y) = \sum_{r_1=1}^{R_1}\sum_{r_2=1}^{R_2}\sum_{r_3=1}^{R_3} \tilde{I}(\boldsymbol{u}_{r_1}^{(1)\mathrm{T}}\boldsymbol{z}_{r_2r_3}, y). \quad (3)$$

其中: $\boldsymbol{u}_{r_1}^{(1)}$ 为矩阵 $\boldsymbol{U}^{(1)}$ 的第 r_1 列, $\boldsymbol{z}_{r_2r_3}$ 为矩阵 $\boldsymbol{Z}_{(1)}$ 的第 r_2r_3 列, 从而将 $\tilde{I}$ 转化为传统形式, 即针对随机变量的求解. 对 $\boldsymbol{U}^{(1)}$ 的求解在转化为传统形式后, 每列的值可以通过负熵的方法对其进行近似并按照梯度上升的方法进行求解[10], 这里不再赘述. 转换矩阵 $\boldsymbol{U}^{(2)}, \boldsymbol{U}^{(3)}$ 也可依次求解, 具体过程如下:

输入: K 个训练数据 $\{\boldsymbol{\mathcal{X}}_k \in \boldsymbol{R}^{I_1\times I_2\times\cdots\times I_n}\}$.

参数: 每个维度的特征数量 $\{R_1, R_2, \cdots, R_n\}$.

过程: 初始化 $\boldsymbol{U}^{(n)} \in \boldsymbol{R}^{I_n\times R_n}$.

Repeat For $n = 1$ to N,

$$\boldsymbol{\mathcal{Z}}_k^{-n} = \boldsymbol{\mathcal{X}}_k \times_{-n} \{\boldsymbol{U}\}^{\mathrm{T}},$$

$$\boldsymbol{U}^{(n)} = \underset{\boldsymbol{U}^{(n)\mathrm{T}}\boldsymbol{U}^{(n)}=\boldsymbol{I}}{\arg\max}\ \tilde{I}(\boldsymbol{F}_{(n)}, y),$$

End

Until {收敛}.

在上述过程中: $\times_{-n}$ 为张量与转换矩阵集合中除第 n 个的转换矩阵的乘积, 结果表示为 $\boldsymbol{\mathcal{Z}}^{-n}$. 收敛条件是 $\Delta I(\boldsymbol{\mathcal{F}}, y)$ 值停止降低, 即互信息值保持稳定.

该方法的主要问题是不能保证全局最优, 而且在维度选择优化时计算量较大, 但相对而言是可以接受的, 毕竟在大数据集中此学习过程只进行一次.

2　综合相似度算法 SCTA

经过降维后的特征张量 $\boldsymbol{\mathcal{F}}$ 实际上是将路径、节点以及单词表示的 XML 文档降到一个更低维度的张量空间中, 并进行相似度的比较. 在进行相似度计算的过程中, 并不直接使用张量的 Frobenius 距离, 而是采用基于路径的文档相似度计算思想, 因为需要充分考虑文档的结构特点, 称此方法为基于张量的文档相似度计算方法 (SCTA).

将两个待比较的张量 $\boldsymbol{\mathcal{X}}$ 和 $\boldsymbol{\mathcal{Y}}$ 文档的相似度定义如下:

$$\mathrm{sim}(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}) = \frac{1}{R_1} \times \sum_{s=1}^{R_1} \max_{1\leqslant t\leqslant R_1} \mathrm{sim}(\boldsymbol{X}_{s::}, \boldsymbol{Y}_{t::}). \quad (4)$$

其中: $\boldsymbol{X}_{s::}$ 和 $\boldsymbol{Y}_{t::}$ 为两个张量的 slice 矩阵; $\mathrm{sim}(\boldsymbol{X}_{s::}, \boldsymbol{Y}_{t::})$ 为这两个 slice 矩阵所在路径 P_s 和 P_t 在低维空间上的相似度, 其值的计算定义为两个 slice 矩阵向量化后的余弦值, 即

$$\mathrm{sim}(\boldsymbol{X}_{s::}, \boldsymbol{Y}_{t::}) = \frac{\mathrm{vec}(\boldsymbol{X}_{s::}) \cdot \mathrm{vec}(\boldsymbol{Y}_{t::})}{|\mathrm{vec}(\boldsymbol{X}_{s::})||\mathrm{vec}(\boldsymbol{Y}_{t::})|}. \quad (5)$$

上述的计算过程实际上是一种在低维空间上结合内容和结构的相似度计算方法.

3　实验结果及分析

实验数据来自两个数据集: 数据集 1 (DS 1)是将来自 20 个网站的超文本标记语言 (HTML) 经过提取和转换后生成的 XML 文档[12]; 数据集 2 (DS 2) 采用了 Sigmod XML 数据集中的文档, 含有以计算机类期刊目录为主题的异构 XML 文档. 实验采用的编程语言是 Java SDK 5, 运行的硬件环境为 Lenovo Thinkpad X201i, 内存为 2 GB, 酷睿 i3-330M 双核处理器, 主频为 2.13 GHz.

结构与内容相结合的相似度计算方法 (SCSC)[11] 是对于文献 [6] 的改进模型. 首先, 生成对应的频率路径模型, 即带有位置和频率权重的最长公共子串 (PFWLCS), 提取出文本内容的特征词; 然后, 依次从两个数据集随机选取一个文件作为初始聚类中心, 与数据集中其他文件进行相似度比较. 实验中的参数 α 通常根据经验或尝试获得, 但文献 [13] 中的实验结果表明, 对于异构文档集, 取 $\alpha = 0.5$ 时的聚类结果会有较为明显的改善, 而后则趋于平缓, 在实验中该结果也得到了验证. 当然, 该参数的最佳取值与文档集本身的特点有关, 不同的文档集需要做大量的尝试才能得出, 而这在实际中显然是不现实的.

SCTA 方法先对数据集中的文档分别采用张量表示和降维, 维度参数分别取 $\{60, 120, 500\}$ 和 $\{25, 50, 300\}$. 依次从这两个数据集中随机选取一个文件作为初始聚类中心, 使用 SCTA 算法与数据集中的其他文件进行相似度比较. 另外, 针对非线性模型的比较, 实验中采用了模型 Bitcube, 而未使用 SLVM, 这主要是因为各文档集中的文档具有独立性, 即彼此之间没有引用关系的存在.

上述实验都采用最近邻聚类进行聚类处理, 设置聚类阈值为 $T = 0$, 采用 $N = 10$ 折交叉实验方式, 取平均值. 最近邻聚类的一般过程是: 1) 随机选取文档 Z_1 作为中心, 并建立一个类; 2) 另取文档 Z_2, 计算 Z_1 与 Z_2 的距离, 如果其值小于设定的阈值 T, 则 Z_2 归入 Z_1 所在类, 否则 Z_2 自成一个新类; 3) 运用相似度算法分别计算其余文档与这两个文档的距离, 得到最近距离, 如果其值小于设定的阈值 T, 则归入最近距离文档所在的类中, 否则自成一个新类; 4) 重复过程直到数据都处理完全.

最终实验结果如表 1 所示. 表 1 表明, 以 SCSC 算法作为距离计算的聚类效果在这两个数据集中都有不错的表现, 与实际的聚簇个数都很接近. 如果与单纯使用结构路径算法的聚类结果进行比较, 则会发现本次实验中的一些指标要偏低一些, 这是因为 DS 1 数据集中的 XML 文档大多是异构的, 而且其结构具有较大的不平衡性. 在这种情况下, 数据集中的文档结构特征比起其中的元素内容特征要重要得多. 这也说

明参数α的选择对于线性综合算法具有一定的敏感性, 也表明文档的结构与内容并不是独立的. 对于非线性的SCTA方法, 实验结果对于数据集DS 1的聚类效果在主要指标上要好于传统SCSC的聚类效果, 与单纯的路径聚类结果基本持平, 说明其对于同类别中异构文档具有相当的分辨能力. 同时, 对于DS 2的聚类结果与SCSC的结果也相差不大, 验证了SCTA对于内容数据的识别性. 与传统SCSC方法需要手动调整相关参数不同, SCTA将内容和结构数据以其原始结构进行分析和处理, 能够体现出算法的自适应性. Bitcube方法在两个数据集中的综合表现不是很好, 原因可能是其按位异或的操作不能很好地体现相似度的精度, 另外, 结构相似度中忽略了节点的位置信息.

表 1 SCTA 与传统聚类结果比较

方法	数据集	召回率%		准确率%		F 1 测度%		纯度%		聚类个数
		micro	macro	micro	macro	micro	macro	micro	macro	
SCSC	DS 1	80.76	79.35	93.64	91.68	86.72	85.07	93.67	92.33	3.64
	DS 2	88.43	84.36	95.26	92.74	91.72	88.35	85.31	83.25	3.86
SCTA	DS 1	84.15	82.05	95.20	91.19	89.33	86.37	94.60	92.37	3.80
	DS 2	89.41	83.40	95.92	92.10	92.55	87.53	86.57	85.65	3.85
Bitcube	DS 1	76.70	75.33	95.50	93.28	82.07	83.34	93.25	90.53	3.76
	DS 2	80.35	79.58	92.24	90.69	85.75	84.57	80.30	81.06	3.80
PFWLCS	DS 1	83.46	80.44	95.71	91.57	89.17	85.65	93.58	90.32	3.80

4 结 论

结构和内容是XML文档分析的两个重要方面, 可以使用线性和非线性的方法对它们进行综合. 线性方法分别使用不同的模型独立地计算结构和内容相似度, 不考虑两者之间的相关性; 而非线性方法则将内容和结构相似度信息融入在一个模型中, 两者之间的相关性不做独立性假设.

本文给出了一种对XML文档进行张量表示和降维的方法, 并设计了基于张量的综合相似度计算算法SCTA. 该方法较好地考虑了内容和结构特征以及两者之间的相关性, 并通过实验数据表明, SCTA在性能指标方面相比于传统的综合方法具有一定的优势.

参考文献(References)

[1] Omidvar, Amin, Mehdi Garakani, et al. Context based user ranking in forums for expert finding using WordNet dictionary and social network analysis[J]. Information Technology and Management, 2014, 15(1): 51-63.

[2] Aïtelhadj A, Boughanem M, Mezghiche M. Using structural similarity for clustering XML documents[J]. Knowledge and Information Systems, 2012, 32(1): 109-139.

[3] 王桐, 刘大昕. 一种新的混合XML文档聚类方法[J]. 哈尔滨工程大学报, 2007, 28(6): 102-105.
(Wang T, Liu D T. A new hybrid method for XML clustering[J]. J of Harbin Engineering University, 2007, 28(6): 102-105.)

[4] Helmer S, Augsten N, Böhlen M. Measuring structural similarity of semi-structured data based on information theoretic approaches[J]. The VLDB J, 2012, 21(5): 677-702.

[5] Guo Yongming, Chen Dehua, Le Jiagin. Clustering XML documents by combining content and structure[C]. Int Symposium on Information Science and Engineering. Shanghai: IEEE Computer Society, 2008: 583-587.

[6] Tran Tien, Nayak Richi. A progressive clustering algorithm to group the XML data by structural and semantic similarity[J]. Int J of Pattern Recognition and Artificial Intelligence, 2007, 21(4): 1-23.

[7] Madani Amina, Omar Boussaid, Djamel Eddine Zegour. Semi-structured documents mining: A review and comparison[J]. Procedia Computer Science, 2013, 22(2013): 330-339.

[8] Yoon J, Raghavan V, Kerschberg L. Bitcube: Clustering and statistical analysis for xml documents[C]. The 13th Int Conf on Scientific and Statistical Database Management. Virginia: Fairfax, 2001: 158-167.

[9] Nadine, Salah Bourennane. Dimensionality reduction based on tensor modelling for classification methods[J]. IEEE Trans on Geoscience and Remote Sensing, 2009, 47(4): 1123-1131.

[10] Leiva Murillo J M, Artes A Rodriguez. Maximization of mutual information for supervised linear feature extraction[J]. IEEE Trans on Neural Networks, 2007, 18(5): 1433-1441.

[11] Piao Yong, Wang Xiukun. A hybrid method for XML clustering by structure and content[J]. J of Software, 2011, 6(12): 2361-2368.

[12] Leung Hopong, Chung Fulai. XML document clustering using common xpath[C]. Proc of the 2005 Int Workshop on Challenges in Web Information Retrieval and Integration. Washington: IEEE Computer Society, 2005: 91-96.

[13] Tien Tran, Richi Nayak, Peter Bruza. Combining structure and content similarities for XML document clustering[C]. Conferences in Practice and Research in Information Technology. Glenelg: Australian Computer Society, 2008: 219-226.

(责任编辑：闫 妍)