一种基于 k-prototype 的多层次聚类改进算法

李士进,朱跃龙,刘 净

(河海大学计算机及信息工程学院,江苏 南京 210098)

摘要:针对 k-prototype 算法在处理复杂的数据集时,常出现一些纯度不高的簇,影响了聚类质量的问题,提出一种基于 k-prototype 的多层次聚类改进算法,利用属性自动选择的方法将一些纯度不高的簇进行再聚类,以提高聚类质量.以 UCI 标准测试数据集进行实验,实验结果表明,该改进算法能够明显提高混合型数据集的聚类质量,并且在数据约简方面有良好表现.

关键词:聚类;混合数据;多层次聚类;k-prototype 聚类

中图分类号:TP311 文献标识码:A 文章编号:1000-1980(2007)03-0342-06

聚类分析是数据挖掘中一个非常活跃的研究分支,具有广泛的应用前景.聚类方法主要有以下几类[1]:(a)划分方法,如 k-means 算法,CLARANS 算法;(b)层次的方法,如 CURE 和 BIRCH 算法;(c)基于密度的方法,DBSCAN 是其典型代表,另外还有 OPTICS 算法;(d)基于网格的方法,STING 是基于该方法的一个典型例子;(e)基于模型的方法,如一些统计学和神经网络的方法.

上述方法存在以下问题:(a)对于可处理的数据类型存在局限,许多算法只限于处理仅包括数值属性或仅包括类别属性的数据集,对于适合于混合属性数据集的聚类算法却较少;(b)需要确定一些参数,并且这些参数的设置和聚类的结果密切相关,特别是很多算法需预先给出聚类的簇数;(c)对数据分布的适应性,一些算法只能发现球状分布的簇,而无法发现任意形状的簇;(d)其他问题,如高维数据的可伸缩性,对输入顺序的敏感性,可扩展性,对噪声的处理能力等.

本文针对问题(a)和(b)进行研究.由于混合型数据集自身的复杂性,在传统的聚类算法中适合于处理这种数据集的算法较少,而且聚类效果也不佳.另外,聚类簇数的确定一直是聚类分析难以解决的问题.目前的聚类集成算法大多是一种并联式结构,由于需要对聚类成员的聚类结果进行匹配和融合,所以时间复杂度较高;同时确定聚类簇数的问题依然存在,特别是聚类成员的簇数、最终的聚类簇数以及两者之间的关系更是一个难以确定的问题.

本文借鉴多分类器集成技术,以 k-prototype 算法为基础聚类算法,设计了一种多层次的聚类集成算法.该算法适合于混合型数据集,采用了级联式结构,避免了匹配和融合的过程,并且只需给出聚类簇数的一个初步估计值,随着聚类层次的增加对簇数进行自适应调整.本文以 UCI 标准测试数据集进行实验,证明了该算法具有较高的聚类准确率,明显提高了混合型数据集的聚类效果,其时间复杂度较低,具有很好的可扩展性.

1 多层次聚类改进算法

k-means 算法最早由 Mcqueen[2] 提出,但它只针对数值属性的数据集进行聚类,而后 Huang[3-4] 提出k-mode 算法和 k-prototype 算法,推广了 k-means 方法,使之可以对类别属性和混合型属性的数据集进行聚类.但由于一些混合型属性的数据集比较复杂,采用 k-prototype 聚类后,存在一部分纯度不高的簇,削弱了簇的代表性,影响了聚类的质量.对此本文提出了一种基于 k-prototype 的多层次聚类方法,针对这些纯度不高的簇进行再聚类.

收稿日期:2006-09-19

基金项目:水利部"948"资助项目(200517);河海大学科技创新基金资助项目(406097)

作者简介:李士进(1973—),男,江苏姜堰人,副教授,主要从事模式识别及数据挖掘研究.

1.1　k-prototype 算法

设数据集 $X=\{x_1,x_2,\cdots,x_n\}$,n 为数据集 X 的对象数目,用 $A_1,A_2,\cdots,A_m$,m 个属性描述对象 $x_i(1\leqslant i\leqslant n)$,给定一个正整数 k,将数据集 X 划分为 k 个簇. k-prototype 算法处理混合型属性数据集,所以该算法是 k-means 和 k-mode 算法的结合,具体算法如下:(a)选择 k 个原型作为簇初始中心;(b)根据相异度将对象分配到最近的簇,分配后即更新簇的原型;(c)在所有的对象分配到簇以后,重新测试对象和簇原型的相异度,如果发现与该对象最近的原型是其他簇的原型而不是当前簇的原型,则将该对象重定位并更新簇的原型;(d)重复(c)直到整个数据集循环测试一遍没有对象改变簇.

由于 k-prototype 算法处理的是混合型属性数据集,这里需要特别强调一下对象间相异度的度量.

设对象 $X,Y\in X$ 都有属性 $A_1^r,A_2^r,\cdots,A_p^r,A_{p+1}^c,\cdots,A_m^c$,上标为 r 的是数值型属性,上标为 c 的是分类型属性,X 和 Y 的相异度如下:

$$d(x,y)=\sum_{j=1}^{p}(x_j-y_j)^2+\gamma\sum_{j=p+1}^{m}\delta(x_j,y_j) \tag{1}$$

其中,$\delta(x_j,y_j)=\begin{cases}0,(x_j=y_j)\\1,(x_j\neq y_j)\end{cases}$. 式(1)的第一项为数值属性的欧几里得平方距离,第二项为类别属性上的简单的匹配相异度,参数 γ 是为了避免偏向于数值或类别属性.

1.2　基于 k-prototype 的多层次聚类算法

近来有文献提出了分类树和聚类树的概念,例如,多层次支持向量机分类树[5]、分级递减聚类算法[6]、属性均值聚类二叉树[7]等,均体现了一种多级、多层次的聚类策略. 这些方法中多层次聚类策略都是一种二叉树结构,而且对于停止再聚类要求过于苛刻,实际数据集较难实现,并且二叉树的层次比较高. 本文提出的一种多叉树结构减少了树的层次,并且通过簇的纯度控制簇再聚类,同时对每个层次的聚类进行自适应属性选择,利用数据集的特性使得再次聚类获得的簇更接近纯度要求.

对于一些复杂的混合型属性数据集,采用 k-prototype 聚类后会出现一些纯度不高的簇,所谓纯度不高就是簇中出现了 2 种或 2 种类别以上的数据,并且各种类别的数据在数目上不相上下,无法体现出簇中某个类别数据的优势,削弱了簇的代表性,影响了聚类质量. 本文提出利用多层次聚类的策略,以 k-prototype 为基础聚类算法,在不同层次上对纯度不高的簇进行再聚类,以使得混杂的各类别数据能够分割开.

1.2.1　纯度阈值的设定和多叉树的结构

设数据集 X 有 n 种类别的对象,簇 $C=\{l_1,l_2,\cdots,l_n\}$,l_i 为簇 C 中类别为 i 的对象集合,簇 C 的纯度定义为 $\max(|l_1|,|l_2|,\cdots,|l_n|)/|c|$,其中 $|l_i|$ 为簇 C 中类别为 i 的对象数目,$|c|$ 为簇 C 中对象的数目. 纯度越高,簇中某一类别的数据在整个簇中所占的比例越大,混杂在该簇中其他类别的数据比例则越小,这个类别越能代表该簇. 如果得到的纯度较高的簇数目越多,聚类的质量就越好. 显然,作为一个理想的簇,其纯度应该为 1,由于实际的很多数据集是比较复杂的,并且存在噪声数据,所以经过一次聚类以后能得到纯度为 1 的簇的数目是很少的. 可以根据实际需要设定一个纯度阈值,纯度小于该阈值的簇需要进行下一层次的再聚类,最后得到更多纯度较高的簇来提高聚类质量.

可以将聚类的二叉树结构扩展成多叉树,对于不同的数据集,根据其类别的特性,可以在第一层次的聚类上设定一个合适的簇数 $k(k\geqslant 2)$,经过一次 k-prototype 聚类后得到 k 个簇,对小于纯度阈值的簇进行下一层次的再聚类. 再聚类时簇数的设定可根据该簇中各类别数据所占的比例来决定,最简单的方法可以是若混杂有 n 个类别的数据就聚成 n 个簇. 相对于二叉树的结构,这样得到的聚类树各结点的子结点数目各个簇都不相同,也即一种多叉树结构.

1.2.2　再聚类时属性的选择

再聚类时如何进行属性的选择是基于对每个属性的分析,对数值型属性和分类属性应该分别进行.

对于数值属性的选择采用如下策略:设簇 $C=\{l_1,l_2,\cdots,l_q\}$,$q\leqslant n$,l_i 为簇 C 中类别为 i 的对象集合;将 l_i 的对象在每个数值属性 $(A_1^r,A_2^r,\cdots,A_p^r)$ 上进行统计分析,分别得到 l_i 的对象在这些属性上的期望和标准差 $\{(e_{i,1},d_{i,1}),(e_{i,1},d_{i,1}),\cdots,(e_{i,p},d_{i,p})\}$.

a. 通过期望来判断在该属性上是否存在某个类别的期望与其他类别的期望相差很大,对于簇 C 判断属性 A_j 是否参与再聚类,可以设定一个系数 $\alpha(0<\alpha<1)$,对于集合 $E=\{e_{1,j},e_{2,j},\cdots,e_{q,j}\}$,令 $e_{\max}=\max E$,如

果存在一个 $e_{i,j} \in E$,使得 $e_{i,j} \leqslant \alpha * e_{\max}$,那么该属性参与再聚类.

b. 通过期望和标准差来判断各类别在该属性上是否有不相交的区间,对于簇 C 判断属性 A_j 是否参与再聚类,观察这些区间$(e_{1,j}-d_{1,j}, e_{1,j}+d_{1j})$,$(e_{2,j}-d_{2,j}, e_{2,j}+d_{2j})$,…,$(e_{q,j}-d_{q,j}, e_{q,j}+d_{q,j})$是否不相交,如果是,那么属性 A_j 参与再聚类.

对于每个分类属性$(A_{p+1}^c, \cdots, A_m^c)$,分别计算 l_i 的对象在这些属性上的信息增益[8] $(g_{i,p+1}, g_{i,p+2}, \cdots, g_{i,m})$. 将这些信息增益从大到小排序,设定一个参数 $\lambda(0 \leqslant \lambda \leqslant m-p)$,可以从前 λ 个信息增益中选择非零值项所对应的类属性参与下一层次再聚类.

1.2.3 算法

综上所述,基于 k-prototype 的多层次聚类算法的具体描述如下:

a. 设定第一层次的聚类簇数 k,纯度阈值以及聚类树的层次限定.

b. 采用 k-prototype 算法对整个数据集进行聚类,得到第一层次的 k 个簇 $\{C_1, C_2, \cdots, C_k\}$.

c. 分析该层中簇 C_i 的纯度,若簇 C_i 纯度大于或等于纯度阈值,则簇 C_i 停止再聚类形成聚类树的叶子结点,否则进行再聚类.

d. 对需进行再聚类的簇 C_i,利用上述方法对数值属性和类属性分别进行分析,选择参与再聚类的属性子集,并按簇中各类别数据所占比例来决定再聚类的簇数 k_i.

e. 根据选择的属性以及再聚类簇数 k_i,利用 k-prototype 算法对簇 C_i 进行再聚类,得到簇 C_i 的 k_i 个子簇 $\{C_{i,1}, C_{i,2}, \cdots, C_{i,k_i}\}$,即完成了簇 C_i 的再聚类. 对该层中其他的簇重复进行步骤 c,d,e 的操作.

f. 当对该层次中所有需进行再聚类的簇完成了再聚类以后,聚类树的层次就增加了一层,此时判断这个聚类多叉树是否已经生长到限定的层次,如果是,则停止下一层次的聚类,这一层次中所有子簇都将成为聚类树的叶子结点. 否则的话,对所有得到的子簇按照上述方法进行下一层次的再聚类.

按照上述算法,可得到一棵聚类多叉树,多叉树的叶子结点即为最终得到的簇. 显然最终得到的簇数大于或等于 k,并且和各参数的设定以及树的层次限定有关. 第一层次的初始聚类簇数 k 只是一个初步估计,在实际应用中可以从数据的目标类别数开始. 另外,由于各种混合属性的数据集特点不同,在采用上述方法进行聚类时,要根据实际应用需要进行参数的设定.

2 实验比较与分析

本文采用 UCI 机器学习数据库[9] 中的 adult 数据集进行实验,该数据集是从美国人口局 1994 年人口普查数据中提取出来的,包含有 6 个数值型属性和 8 个类别型属性,其目标类别是指示一个人的收入是否超过 $50 000. 该数据集总共有 48 842 条记录,以 2∶1 的比例将记录划分为训练集和测试集,再去除含有缺失值的记录,最终训练集有 30 162 条记录,测试集有 15 060 条记录. 实验硬件环境:CPU 为 INTEL PIII 930,内存为 256M. 采用C++编程语言,Visual C++6.0 环境下编译运行.

2.1 评判指标

设数据集 D,其目标类别有 n 类,假设聚类得到 k 个簇 $C_1, C_2, \cdots, C_k$,其中 $C_i = \{l_{i,1}, l_{i,2}, \cdots, l_{i,n}\}$,$l_{i,j}$ 为簇 C_i 中的第 j 类目标类别的样例集合,为评价数据集 X 的整体聚类质量,定义了聚类准确率

$$r = \frac{\sum_{i=1}^{k} a_i}{|D|} \qquad a_i = \max(|l_{i,1}|, |l_{i,2}|, \cdots, |l_{i,n}|) \tag{2}$$

即簇 C_i 中数目最多的那一类样例所对应的目标类别就是该簇的类别.

聚类最终结果是对数据集的一个划分,评价聚类的效果可以对该划分作一个评价. 本文以训练集聚类的结果,也就是这个训练集上的划分作为分类的依据,对测试集做了最小距离分类,用聚类的结果来预测数据对象的类别. 在数据挖掘应用中,有时为了对数据进行预处理,需要在挖掘之前进行数据约简(data abstraction). 本文以测试集的分类错分率来评价聚类结果的好坏. 测试集的分类错分率定义为

$$\text{分类错分率} = \frac{\text{预测错误的样例数}}{\text{测试集的总样例数}} \tag{3}$$

显然,错分率越低,聚类质量越好.

2.2　实验的对比算法

为了验证多层次聚类集成算法的性能,与 2 种已经报道的典型聚类集成算法进行对比实验,它们分别是 基于初始簇中心的选择性聚类集成算法[10]和基于不同类型属性的聚类集成算法[11]. 第一种方法以不同的初始簇中心利用 k-prototype 算法产生聚类成员,任意选择 t 个基于投票的方法进行选择性聚类集成. 第二种将数值型聚类算法和类别型聚类算法进行集成,来解决混合型数据集的聚类问题,这种方法也体现了一种集成聚类的思想,它选用不同的特征子集来投影,得到不同的聚类成员,然后再将聚类成员的聚类结果进行合并,对合并后的新数据集进行聚类,得到最终的聚类结果.

2.3　聚类准确率比较实验

对于多层次的聚类集成算法,在 adult 训练集上分别建立了 1 层聚类树、2 层聚类树和 3 层聚类树,其中 1 层聚类树实际就是将该算法退化为单纯的 k-prototype 算法,记录 10 次实验的结果,见表 1.

第一层次的聚类簇数 k 设置为 10,观察 2 层聚类树的 10 次实验,其最终得到的簇数都在 10 ~ 20 之间,同样也观察 3 层聚类树的 10 次实验其最终得到的簇数都在 20 ~ 40 之间. 为了进行公平比较,对于基于初始簇中心的选择性聚类集成算法,分别将聚类成员的簇数设为 15 和 30 进行了 2 组实验,在每组实验中都进行集成规模 $t = 5$, $t = 10$ 和 $t = 15$ 的聚类集成,具体的实验结果见表 2[10]. 对于基于不同类型属性的聚类集成算法,其 2 个成员聚类器的簇数 $k = 20$,最终的聚类簇数也设为 15 和 30 进行了 2 组实验,具体的实验结果见表 3[11].

比较表 1 和表 2 的实验结果,发现在 2 组对比实验中,多层次聚类集成算法的聚类准确率都要比基于初始簇中心的选择性聚类集成算法高,并且随着层次的增加,前者聚类准确率提高很快,而后者随着集成规模的增加,其聚类准确率却提高很慢,虽然对于后者可以寄希望于通过增加集成规模 t 来达到 3 层聚类树的聚类准确率,但是由于基于初始簇中心的选择性聚类集成算法需要对聚类成员的聚类结果进行匹配和融合,因此随着集成规模的增加,其时间复杂度将变得很高. 实验时也发现基于不同类型属性的聚类集成算法,很难确定其各聚类成员的聚类簇数和最终的聚类簇数之间的关系.

表 1　多层次聚类集成算法结果

Table 1　Results of multi-level clustering algorithm

实验序号	实验准确率		
	1 层聚类树	2 层聚类树	3 层聚类树
1	0.812 678	0.817 055	0.887 938
2	0.801 339	0.821 166	0.901 664
3	0.791 426	0.818 580	0.889 165
4	0.797 792	0.815 728	0.884 026
5	0.811 186	0.816 955	0.915 191
6	0.814 734	0.817 618	0.851 90
7	0.794 609	0.816 126	0.902 725
8	0.797 063	0.816 756	0.903 886
9	0.809 628	0.818 480	0.949 804
10	0.812 612	0.817 618	0.897 255
平均值	0.804 307	0.817 608	0.898 360

表 2　基于初始簇中心的聚类集成算法结果

Table 2　Results of clustering algorithm based on random selection of initial center of clusters

实验序号	$k = 15$ 时的聚类准确率			$k = 30$ 时的聚类准确率		
	$t = 5$	$t = 10$	$t = 15$	$t = 5$	$t = 10$	$t = 15$
1	0.803 395	0.811 639	0.813 892	0.813 492	0.823 759	0.827 605
2	0.806 346	0.805 513	0.821 936	0.809 736	0.819 374	0.827 630
3	0.804 522	0.811 639	0.817 738	0.810 093	0.825 47	0.823 375
4	0.803 428	0.800 919	0.807 761	0.810 543	0.829 542	0.832 019
5	0.804 390	0.822 450	0.809 205	0.816 370	0.817 603	0.825 309
6	0.804 953	0.813 170	0.814 953	0.812 934	0.813 908	0.829 945
7	0.812 413	0.819 760	0.812 413	0.808 460	0.825 021	0.837 644
8	0.808 633	0.807 044	0.821 936	0.813 875	0.827 765	0.828 845
9	0.799 649	0.819 296	0.819 296	0.815 902	0.818 065	0.827 930
10	0.806 213	0.806 419	0.826 419	0.807 732	0.820 378	0.831 278
平均值	0.805 395	0.811 151	0.816 555	0.811 914	0.822 089	0.829 158

由表 1 可以发现,多层次聚类集成算法随着聚类树层次的增加,其聚类准确率也提高了,这是因为随着层次的增加,表明在上一层中一些纯度不高的簇又进行了下一层次的再聚类,使得这些簇的聚类质量得到提高,这一点再次证明了进行属性选择的再聚类方法的有效性,特别是在 3 层聚类树的时候,聚类准确率几乎接近纯度阈值,体现了多层次聚类集成算法优良的聚类效果.

2.4　分类预测能力对比实验

利用多层次聚类集成算法经过聚类后将训练集分割为若干簇,每个簇都有各自的簇原型和标识,对于测试集中的每个对象,其预测类别即为这若干个簇中与其距离最小的簇原型所对应的簇标识.

利用上述实验得到的 2 层聚类树和 3 层聚类树的聚类结果,对 adult 的测试集进行最小距离分类预测,并与在原始数据集上的 k 近邻算法(1-近邻,3-近邻)分类结果进行比较,错分率的统计如表 4 所示. 2 层聚类

树得到的平均错分率为0.1772,3层聚类树的平均错分率只有0.1682,而1-近邻算法的错分率为0.2142,3-近邻算法的错分率为0.2035,显然,聚类树的错分率均低于k近邻算法的错分率.可以看出,通过该算法聚类得到的簇原型能够很好地代表原来的数据,表明聚类效果较好.

表3 基于不同类型属性的聚类集成算法结果

Table 3 Results of clustering algorithm for clusters with different attributes

实验序号	聚类准确率	
	$k=15$	$k=30$
1	0.798256	0.813265
2	0.797659	0.805528
3	0.796764	0.807205
4	0.806810	0.813524
5	0.796272	0.805510
6	0.803925	0.816681
7	0.793913	0.809251
8	0.802666	0.808241
9	0.808899	0.814254
10	0.796698	0.812195
平均值	0.800186	0.810566

表4 测试集最小距离错分率

Table 4 Misclassification rate of minimum distance classifier of test data set

实验序号	分类错分率	
	2层聚类树	3层聚类树
1	0.1742	0.1702
2	0.1734	0.1665
3	0.1801	0.1793
4	0.1706	0.1696
5	0.1843	0.1602
6	0.1852	0.1724
7	0.1713	0.1675
8	0.1796	0.1593
9	0.1804	0.1655
10	0.1722	0.1712
平均值	0.1772	0.1682

2.5 实验小结

将本文设计的多层次聚类集成算法分别与基于初始簇中心的聚类集成算法和基于不同类型属性的聚类集成算法进行了对比实验,这3种聚类集成算法都适合于处理混合型数据集,从聚类效果上看,多层次聚类集成算法均优于后两者,该算法能够明显提高混合型数据集的聚类质量;在时间效率上,该算法明显高于基于初始簇中心的聚类集成算法,和基于不同类型属性的聚类集成算法相差无几,但是后两者都需要确定聚类成员的聚类簇数和最终的聚类簇数,而本文算法只需给出一个初步的估计值,特别是基于不同类型属性的聚类集成算法很难确定聚类成员的聚类簇数和最终的聚类簇数之间的关系.另外,多层次聚类集成算法还表现出了较好的分类预测能力,其分类错分率明显低于原始数据集上的k-近邻算法.

3 结 语

k-prototype算法是混合型属性数据集最常用的聚类算法之一,但在处理一些复杂的数据集时,常出现一些纯度不高的簇,影响了聚类质量.本文提出了一种基于k-prototype的多层次聚类算法,利用属性自动选择的方法将一些纯度不高的簇进行再聚类,以提高聚类质量.

实验结果表明,本文提出的方法能够明显提高混合型数据集的聚类质量,并且在数据约简方面也有很好的表现.

参考文献:

[1] HAN J,KAMBER M.数据挖掘概念与技术[M].范明,孟小峰,译.北京:机械工业出版社,2001:231-232.

[2] MCQUEEN J. Some methods of classification and analysis of multivariate observations[C]//Proc 5th Symposium on Mathematics & Statistical Probability. Berkeley:Univ of California,1967:281-296.

[3] HUANG Z. A fast clustreing algorithm to cluster very large categorical data sets in data mining[C]//Proceedings of the SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery. Arizona:ACM Press,1997:1-8.

[4] HUANG Z. Extensions to the k-means algorithm for clustering large data sets with categorical values[J]. Data Mining and Knowledge Discovery,1998(2):283-304.

[5] 张国宣,孔锐.基于核聚类方法的多层次支持向量机分类树[J].控制与决策,2004,19(11):1305-1307.

[6] 王连亮,陈怀新.基于改进的模糊c-均值的分级递减聚类算法[J].系统工程与电子技术,2005,27(7):1304-1306.

[7] 贺仁亚,程乾生,孙喜晨.属性均值聚类二叉树及其在人脸识别中的应用[J].北京大学学报:自然科学版,2002,38(5):616-

621.
[8] QUINLAN J. Induction of decision trees[J]. Machine Learning, 1986(1): 81-106.
[9] UCI Repository of machine learning databases [EB/OL]. Berkeley: Department of Information and Computer Science, University of California, [2006-03-10]. http://www.ics.uci.edu/~mlearn/MLRepository.html.
[10] 唐伟，周志华. 基于 Bagging 的选择性聚类集成[J]. 软件学报，2005，16(4)：496-502.
[11] HE Z, XU X, DENG S. A cluster ensemble method for clustering categorical data[J]. Information Fusion (IF '05), 2005, 6(2): 143-151.

An improved multi-level clustering algorithm based on k-prototype

LI Shi-jin, ZHU Yue-long, LIU Jing

(*College of Computer and Information Engineering, Hohai University, Nanjing* 210098, *China*)

Abstract: When k-prototype algorithm was employed to process complex data sets, some clusters with low-purity always occurred. An improved multi-level clustering algorithm based on k-prototype was proposed to tackle the above problem. In order to improve the quality of clustering, re-clustering was performed on those clusters with low-purity through automatic selection of attributes. Experimental results on data set from UCI machine learning repository show that the present algorithm can improve the clustering quality evidently, and it is also suitable for data abstraction.

Key words: clustering, mixed data, multi-level clustering, k-prototype clustering