论文题目：Logical-Shapelets: An Expressive Primitive for Time Series Classification

**作者：盛锦超 1032012026064 电子信息**

**1.陌生词汇**

an order of magnitude一个数量级

intelligent caching and reuse of computations 计算缓存与重用

the admissible pruning of the search space 允许搜索空间的剪枝

gesture recognition 手势识别

learn the inherent structure of the data in a manner that allows intuitive interpretation 以一种允许直观解释的方式学习数据的内在结构

nonuniformly sampled 非均匀采样

untenable 站不住脚根的

a pathological case 一记病例

configuration 外型

monotonically 单调的

an atomic operation 一次原子操作

extraordinary 非凡的

recursively 递归地

have taken extraordinary lengths 付出了巨大的努力

perpetuity 永恒

amenable 有责任的

**2.本文介绍**

**ABSTRACT：**

Shapelets是一条或者一类时间序列中具有最显著特点的子序列，这个想法是很自然的，我们知道时间序列存在着多维和大量噪声点等情况，如果我们将整条时间序列进行处理，不仅复杂度非常高，而且会存在着准确率和质量低下的情况；例如在鸟叫的时间序列里，鸟在一定时间内不可能一直在叫，我们只需要找到那一段叫声的时间序列表示就可以分出不同类别的鸟了(前提是鸟叫声可以通过时间序列得到有效区分)；Shapelet的计算通过ED而不是DTW，ED是欧式距离；很显然我们需要局部特征的比较而不是DTW这类可以将不同长度的时间序列进行距离比较

Shapelets具有巨大的潜力但是同样存在缺点首先他们的表现力体现在了二元分类上，但不可否认计算量和计算时间同样也是相当大的

Logical-Shapelets解决后者问题，能够在更短的时间内发现它；这也是本文的目标，且本文的思想是在暴力法的基础上探索出来的，我们会先介绍加速算法，再提出这个观点

**INTRODUCTION：**

Shapelets是解决二分类问题的，当有多类的时候，可以使用决策树进行分割

Shapelets受欢迎的原因在于首先，在许多情况下，时间序列shapelets能够以一种允许直观解释的方式了解数据的内在结构；其次，shapelets通常比原始的时间序列要短得多，并且与基于实例的方法不同，这些方法需要与整个数据集进行比较，我们在分类时只需要一个Shapelet；但是shapelets的当前定义不足以表达某些在现实世界中很常见的概念且计算非常耗时，对数据要求很高，不能很好在有噪声，非均匀采样的情况下执行

新算法首先计算统计量，然后我们使用一种新的允许剪枝技术来跳过对绝大多数候选形状的代价高昂的熵计算；我们进一步说明，我们可以在逻辑表达式中组合多个Shapelets，从而可以描述复杂的概念

需要注意的是，逻辑shapelets的表达性本质上是零成本的。如果我们将它们应用于不需要其增强的表示能力的数据集，它们就会退化为经典的shapelets，这是一种特殊情况

**DEFINITION AND BACKGROUND：**

为了实现尺度和偏移不变性，在计算实际距离之前，我们必须使用z归一化对单个时间序列进行归一化；此外还必须长度标准化；公式为

定义时间序列，定义熵，定义衡量质量的两个标准信息增益和分离间隙，定义shapelet；这类定义在文中会有具体介绍

介绍原始暴力算法，存在着两个标准化计算存在计算冗余的问题，这样会导致计算复杂度增加，时间复杂度为O(N2m4)显然很大，故我们需要降低复杂度，重点应该在m上即子序列上

暴力法的思维是很简单的，每条时间序列规定长短的连续值计算与其他时间序列的大小，这里一般用ED求最小，然后得到orderline，再通过一个Γ值分出左右两部分，然后得到信息增益和gap值，最后选取信息增益最大的最为shapelet候选，具体内容在算法1

**SPEEDUP TECHNIQUES**

本节介绍如何加速：

第一是有效距离计算；通过计算冗余问题设置五个数组量(具体内容见论文介绍)，存放平均值和标准差和一个偏差计算的数组，就可以计算出来子序列与原始序列之间的距离；算法至少省略了一个O(m)的计算量

第二是候选剪枝；思路如果知道一个子序列是一个比较差的子序列，那么我们将其相似的子序列直接过滤，就比如在过滤Si,l和Si+1,l的时候，显然如果前者是非常差的，那么后者大概率也是非常差的，过滤是符合常理的；下面直接给出一结论：将一个点从它的小分区转移到它的主分区总是会增加信息增益，意思是当分为两类的时候，将少数的放到多数的情况下，必然会增加信息增益；在两个类中，不同的转换方向并不会造成大分区的改变，而多个类情况下则不能保证，好在我们许多数据集的类都不大，我们通过2的c次方把所有结果计算出来；总结是两类的计算是准确的，而多类是启发式与经验式的探索不同的情况进行计算；于是我们可以设置上界，当一条子序列小于这个上界了，直接过滤而不计算结果；由于上界计算是基于三角不等式的，因此我们只允许使用与当前候选序列相同长度的序列之前计算的orderlines；

描述最终算法4；每次for循环先算这条时间序列和其他时间序列的五个统计量，然后设立一个H数组，大小这里常用为5；当新的子序列跑进来后，我们先和H里的orderline进行比较，通过算法3判断是否会改变信息增益，如果未发生改变上界，则直接过滤，否则再计算新的orderline再判断是否是最好的shapelet，当然我们需要注意当最终不改变的情况下，就要将该orderline装到H上，原则是最老优先

**LOGICAL SHAPELET**

这部分介绍一个思想，能不能对shapelet进行逻辑上的组合；在可能的情况下，有些时间序列存在着两个及以上最显著的特征来代表该时间序列；逻辑组合最重要的就是and和or，当然还有其他的，为了防止过拟合，我们将阈值只设置成一个且逻辑只选择and和or，and就取最大的shapelet距离，or取最小的shapelet距离

我们修改算法4，当执行了算法4得到shapelet后，我们看看是否有类的例子分到了不同的分区；如果有就再次寻找shapelet使其能够分类，但注意新找的shapelet应该与之前的没有任何交集；就这样一直找直到分类正确，但很容易发现这会产生过拟合问题；在这项工作中，我们设置最多4个，然而，对于一个非常大的数据集，我们可以慢慢地增加这个数字同时检查过拟合；注意，这里的情况几乎完全类似于决策树中的过拟合问题，我们希望类似的解决方案也能在这里工作

**EVALUATION**

本节希望通过实验证明两点，第一是首新加速技术可以用于更快地找到经典的和逻辑的shapelets，就是算法4；其次是证明真实存在我们的Logical shapelet这个思维的问题；通过实验证明我们的算法对所有数据集都获得了一些加速，最大加速值为27.2

**CASE STUDIES**

介绍了三个案例，其中第二个案例是索尼的小狗机器人，它配了一系列传感器，使其在地毯和水泥地上行走

根据测量，水泥地的阻力更大，故时间序列变换浮动更大；实例证明，通过and两个shapelet得到的主准确率最大且耗时也较少

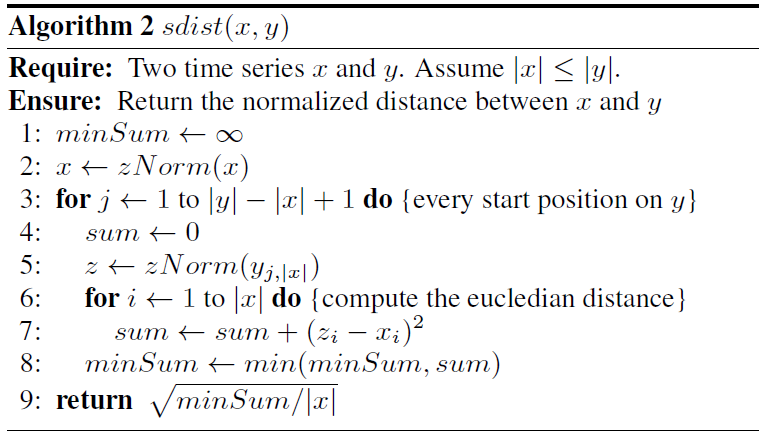
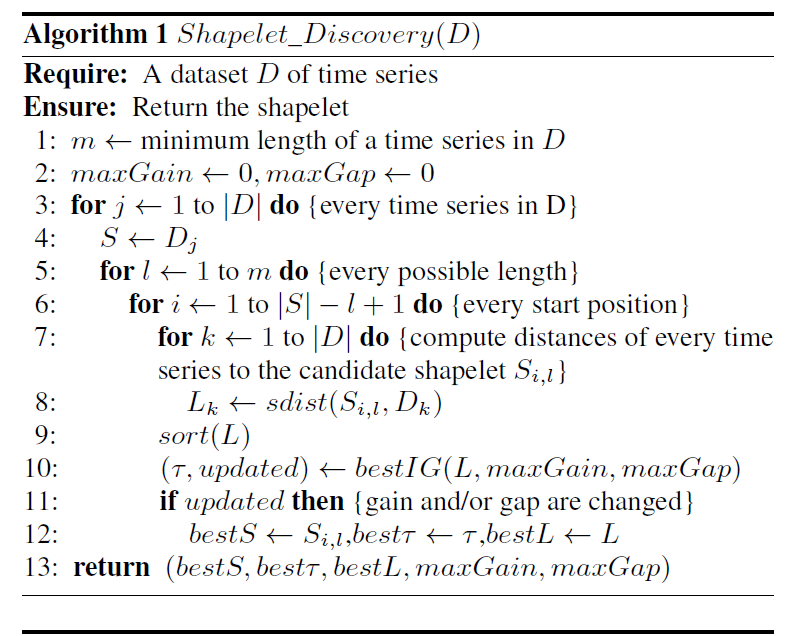
**CONCLUSIONS**

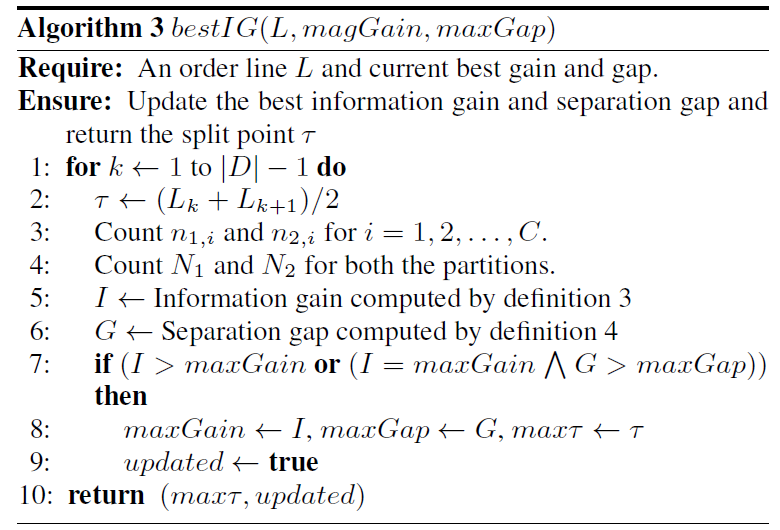
本文介绍了一种新的时间序列分类原语，它比传统的shapelets具有更强的表现力；本文演示了时间序列数据集中逻辑概念的存在，以及逻辑形状函数在手势识别、机器人技术和用户认证等领域中的效用；本文进一步描述新的技术来有效地找到经典的和逻辑的shapelets且通过实验证明方法比其他方法快的多

**3.算法描述**

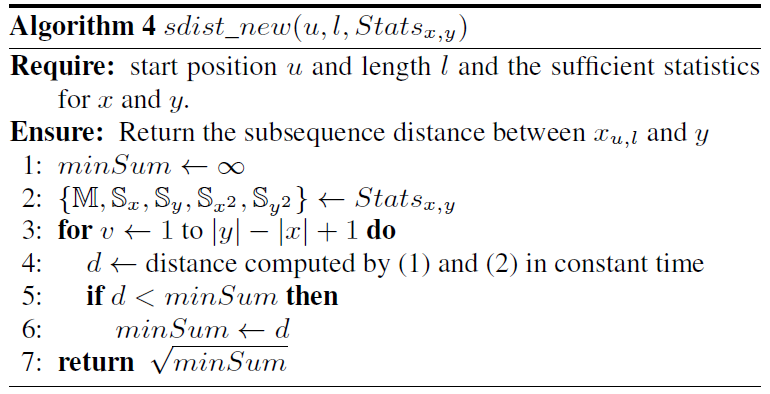
算法1暴力算法

算法数据可能是多类且长度不等

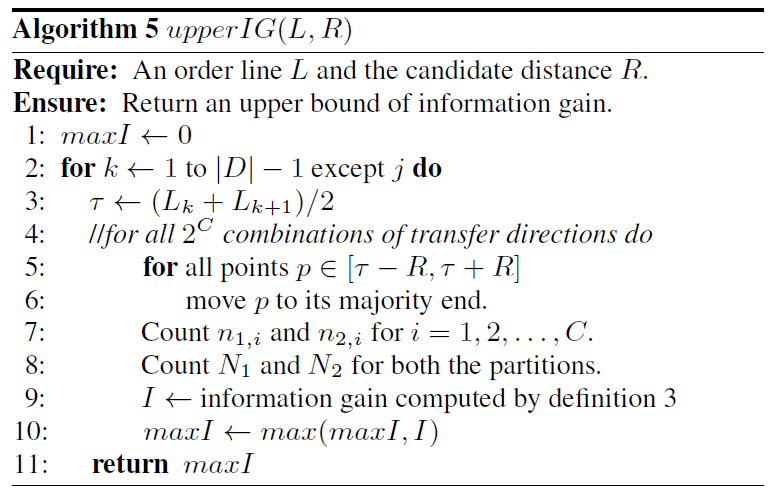




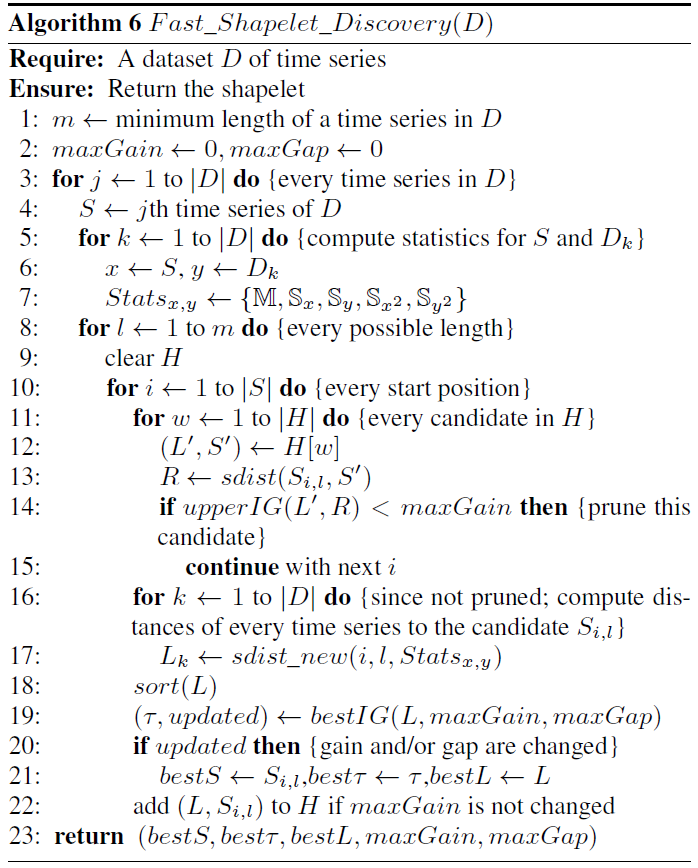
算法2改进距离计算



算法3信息增益上界的计算



算法4最终优化算法



**4.存在问题**

①观察到的最显著的问题是该问题对类数量大的情况下，算法的求解能力还不如暴力法来的快；只能适合少类别且数据量本身也不太大的情况

②逻辑选择部分描述笔者认为是不够清楚的，且文中说可以通过一些简单的修改再找下一个最适合的shapelet，笔者认为不管是从原本算法的执行难度和算法的执行流程来说，都没有说清楚，这点是蛮遗憾的；从笔者的视角里，我们可以将该上条shapelet的有关子序列给清除掉，然后再次搜索，这里可以给个判断标记

③在构建文章最难的那一部分，就是第二个加速问题的适合，文章虽然证明了从小分区到大分区信息增益会增加，但是为什么一定是从小分区到大分区，或者实际上在那个阈值内的所有dis如何进行跳跃的内容文章描述的也不够特别详细，虽然文章从后面说明了二类和多类问题，跳跃的影响是有2的c次方的可能性的，但是仍旧没有回答会如何进行跳跃，可能笔者知识水平不足，不能够理解这个可能比较显然的问题，但笔者认为说明一下会更好吧

④笔者在阅读五个统计量的时候，曾经观察过其他英文和中文的文献对平方的统计量有不同的说明，笔者思考为什么有些统计量会直接平方相减而有些作者是加括号相减再平方，这也是笔者思考疑惑点之一，希望通过阅读更多的文献加以理解

**5.附录说明**

文章作者：Abdullah Mueen

文章、代码和数据集下载：http://alumni.cs.ucr.edu/~mueen/LogicalShapelet/