实现自适应空间文本分割树

王梓涵　周昕逸　刘权

1　理论介绍

在日常的信息传播中，常常需要处理用户所提供的一系列包含空间和关键词的查询项。例如：一些基于位置信息的推荐系统可根据用户的需求推送相关产品的信息，Twitter 等社交网络可分析用户发布的内容及附带的地理位置标签推送他们感兴趣的内容。本文实现的自适应空间文本分割树（Adaptive spatial-textual Partition Tree，以下简称 AP 树）可以通过自动计算关键词和空间因子两种分割方式的成本选择合适的匹配方式。

AP 树的实现有三大挑战。第一，在实际应用中的数据量极大，算法效率的提升可以节省很多成本。第二，算法吞吐量要大以应对源源不断的包含空间文本信息的对象流。第三，需要建立新的空间因子和关键词的索引机制用于不同的分配方法。

AP 树的基本运算包含三个部分。匹配对象和查询项的算法，计算关键词和空间因子分割成本的模型和建立索引的算法。

AP 树包含三种类型的结点：关键词结点（k-node），空间结点（s-node）和查询项结点（q-node）。一棵 AP 树的叶子结点都是 q-node，每一个查询项都可根据其关键词和空间因子被分配入一个或多个叶子结点中。运用关键词分割方法的结点称为 k-node。我们认为所有的关键词都包含于一个有序的词典中，它们的相对位置是确定的，因此可以比较大小。每一层的 k-node 中包含了其叶子结点对应的查询项的第结点层次个关键词的划分（cuts）。运用空间面积分割方法的结点称为 s-node。每一层的 s-node 中包含了其叶子结点对应的查询项的面积划分（cells）。如果一个查询项没有足够的关键词，无法寻找到一个对应的 cut，就用一个 dummy cut 保存；如果一个查询项的覆盖面积已经包含一个 s-node 的覆盖面积，就用一个 dummy cell 保存。

在匹配对象时，使用深度优先搜索法。如果当前结点是 q-node，则判断该对象是否与 q-node 中的查询项匹配，如果匹配则可等待输出。如果当前结点是 s-node，则访问它的 cell 和 dummy cell。如果当前结点是 k-node，则访问它的 cut 和 dummy cut。

AP 树还提供了一个计算两种分割匹配算法的成本的模型。匹配成本 $C(P)$ 的定义为：

$$C(P) = \sum_{i=1}^{f} \mathbf{w}(B_i) \times \mathbf{p}(B_i)$$

其中 B 是一种分割方式，$\mathbf{w}(B)$ 是与 B 有关的查询项的个数，$\mathbf{p}(B)$ 是 B 的击中概率，即在对象匹配过程中 B 被访问的可能性。在关键词分割法中，$\mathbf{p}(B) = \sum_{w \in B} \mathbf{p}(w)$，其中 $\mathbf{p}(w) = \frac{freq(w)}{\sum_{w \in P} freq(w)}$，

$freq(w)$ 为关键词 w 在所有查询项中出现的频率。在空间分割法中 $\mathbf{p}(B) = \frac{Area(B)}{Area(N)}$，$Area(B)$ 是 B 的面积，$Area(N)$ 是结点 N 的区域面积。接着使用启发式算法寻找使得匹配成本最小的关键词的分割和空间的分割。

AP 树还提供了建立索引的方法。使用两个标志 kP 和 sP 来表示所有的查询项是否可以进一步地进行关键词的分割和空间的分割，并通过计算两种分割方式的成本 C_k 和 C_s 确定当前结点使用哪种分割方式。

2 代码实现

2.1 接口设计

2.2 存储实现

2.3 运算实现

3 性能测试

4 评价

5 未来工作

A 源代码

B 数据集来源

C 分工