旅游路线分层聚类与去重方案建议

相关研究与思路

已有研究多从路线规划或序列模式挖掘角度考虑旅游路线聚类。Lee 等（2019）提出使用顺序模式挖掘算法对游客行程进行聚类，首先构建景点出现频率矩阵并剔除低频模式，再根据频繁序列聚类相似或完全相同的路线。Vathis 等（2023）在度假规划问题（VPP）中采用多层次聚类（multi-level clustering）方法，将景点、交通等逐级聚合，以辅助生成多日行程。类似地，Zhou 等（2022）设计了改进的AGNES算法，结合POI的属性和空间距离对景点进行层次聚类，以获得属性相关度高的景点簇。以上工作表明，可通过分层聚类或序列挖掘方法将相似行程归为簇，进而减少冗余，优化旅游产品规划。

特征工程建议

针对旅游路线数据，可提取以下常用特征：

POI身份/类别：直接使用景点、酒店、餐厅等POI的ID或类别编码。Huang 等（2019）将POI ID和类别映射到向量嵌入作为输入特征。类别信息（如“景点”、“酒店”）能体现路线组合特征，对分簇很重要。

空间属性：利用POI坐标计算路线的空间形态和距离特征，如起止点间距离、路线覆盖区域大小等。Zhou 等的聚类方法即将空间距离纳入目标函数。

序列/时间顺序：考虑POI访问顺序和时间间隔，可采用时序编码或动态时间规整（DTW）度量路线相似度。Huang 等指出位置（顺序）编码对模型性能影响较大，即考虑访问顺序能区分不同路线。

打分/偏好等权重：若可用用户对POI的评分或偏好信息，可作为路线优先级指标。高评分POI常被行程优先选择，可用来衡量路线中“重要”景点比例。

结合以上特征，可构建路线向量或序列表示。例如，将每条路线编码为「景点ID序列+酒店序列+时间戳」等，然后设计合适的相似度度量（Jaccard、编辑距离、Frechet、Hausdorff、DTW等）进行聚类计算。Lee 等即利用站点频率矩阵隐式考虑了序列模式。综合考虑POI类别分布、空间布局和时间顺序，有助于反映路线的实际相似性。

聚类算法与实现建议

分层聚类思路：按照题目需求，第一层只考虑景点POI。可先将所有景点列表完全一致的路线归为同簇（即按景点集合或序列精确分组）。这一步类似“哈希聚类”：对每条路线提取景点ID集合或有序列表，以此作为键做字典分组。这样可快速去重和合并相同景点路线。

递进细化：在第一层簇内，再引入酒店、餐厅、交通点等特征做次级聚类。具体可对每个一级簇内的路线，基于酒店序列（或其他类别）属性再次分组。多层次聚类方案中，每增加一类POI，簇进一步细分。Vathis 等即采用此类思路对多个维度逐级聚类。此外，可考虑访问时间顺序，将访问时段作为特征，进一步拆分簇。

算法实现：可采用自底向上（凝聚式）层次聚类算法（如SciPy/Scikit-learn的AgglomerativeClustering）对路线向量进行聚类，从而得到一个树形簇结构；也可用密度聚类（DBSCAN/HDBSCAN）发现任意形状簇，并利用HDBSCAN的层级输出获得树。对于序列数据，还可以用序列聚类方法（如基于最长公共子序列或DTW距离）。Lee 等工作采用顺序模式挖掘分两阶段聚类路线，可作为参考。总之，常见工具如SciPy的 linkage/dendrogram、scikit-learn的 AgglomerativeClustering、HDBSCAN（pip包）等都可用于实现层次聚类。

去重策略：对完全相同的路线（包括全部POI）可直接去重保留一条，同时记录其出现次数。结构化方案可在构建树时，将重复路线视为属于同一叶子簇而非创建新簇。

工具包与可视化

可复用的Python工具包括：

SciPy：scipy.cluster.hierarchy.linkage和dendrogram用于层次聚类及绘制聚类树。

scikit-learn：AgglomerativeClustering、DBSCAN可直接聚类；feature\_extraction等模块可处理类别数据。

HDBSCAN：层次密度聚类库，能输出簇的层级结构，适合发现任意密度簇。

序列相似度：可用dtaidistance或fastdtw计算DTW距离，或python-Levenshtein算序列编辑距离。

图形化工具：Matplotlib/Plotly绘制聚类树（如树状图）；NetworkX/Graphviz或ETE3绘制任意树结构；Folium/GeoPandas可在地图上可视化路线簇。

数据处理：Pandas便于处理路线列表和特征列，GeoPandas或Shapely可处理地理信息（Frechet距离通过Shapely可算）。

这些工具组合可支持从数据处理、聚类计算到结果展示的全流程。

初步设计方案

数据准备：将每条路线表示为结构化列表（如按时间排序的POI对象），并标记每个POI的类别、坐标、评分等属性。

层级聚类流程：第一层仅取景点列表，为每条路线生成“景点签名”（如集合或序列字符串），然后按签名分组，形成若干初级簇；对每个初级簇，检查簇内路线是否仅在顺序或其它POI上有差异。第二层对每个簇内路线引入酒店属性（或餐厅等），再次按新增特征签名分裂子簇。可继续添加更多层（交通、时间窗口等），直至达到设计深度。此过程类似构造一个决策树：每层特征维度决定分裂。

灵活可复用：实现时将聚类层数、使用的POI类型和相似度阈值等参数化，使方案可根据需求调整。比如可配置“仅用景点+酒店”，“用景点+餐厅+交通”等模式；或对某层使用相似度合并（如两个景点集合相似度大于阈值合并）。整个聚类过程可封装为函数或类，方便重复使用。

可视化输出：聚类完成后，用树状图展示层次关系，或用分层嵌套列表/JSON形式输出每个簇的路线列表。在界面上，可在地图中着色标示出不同簇的代表路线，或者用scipy.dendrogram直观展示聚类结果。

去重策略：对重复率高的路线，在第一层就能归为同簇并作去重处理，减少后续计算量。对部分相似但不全同的路线，则落入同簇后层，根据其他属性细分。

本方案模块化且灵活，可在Python中使用上述工具包轻松实现。通过多层特征融合与层次聚类，可有效归并相似路线、保留类别丰富性，并生成清晰的簇树结构。

参考文献

Huang et al., Dynamic Recommendation of POI Sequence… (IJGI 2019)：介绍了POI序列推荐模型及采用的特征（POI嵌入、类别嵌入、地理影响、位置编码）。

Lee & Han, Clustering of Tourist Routes for Individual Tourists… (J Supercomputing 2020)：提出基于顺序模式挖掘的旅游路线聚类方法，分两阶段提取频繁序列并聚类相似路线。

Vathis et al., The Vacation Planning Problem: A multi-level clustering-based metaheuristic approach (Comput. & Oper. Res. 2023)：针对度假规划提出多层次聚类方案，将不同特征层次逐级聚合。

Zhou et al., Tour-Route-Recommendation Based on Improved AGNES Spatial Clustering… (ISPRS IJGI 2022)：构建了结合POI属性与空间距离的层次聚类算法，聚合属性相关度高的景点簇。