路径相似度算法的设计

Yuhong Zhong
Symphony

1 导言

本文旨在给出“跟着电影去旅行”应用中为实现用户推荐路径的功能所使用的计算任意两条路径的相似度的算法。并将分别给出该算法提出的背景及相应需求，算法设计的基本思想，算法的形式化描述，以及在未来可能的改进。

2 算法的背景

在“跟着电影去旅行”应用中，用户可以查询某电影/电视剧的拍摄或取景地点，以及这些地点在电影/电视剧中所对应的片段。进一步，用户可以选择这些地点构建一条路径，作为关于一个或多个电影/电视剧的主题旅游的路径指导。需要特别说明的是，在本应用中所提及的路径，并不是用户进行主题旅游时真正行走的路径，而是主题旅游中需要访问的景点及其先后顺序。

在用户自己构建了一些路径之后，本应用将会根据用户所创建的路径，或是用户所关注的其他路径，或者根据其他信息来为用户推荐其可能感兴趣的其他路径，供用户浏览与关注。在为用户推荐路径时，较为直观的想法即为若两条路径较为相似但又不相同，并且用户关注了其中一条路径，则用户也有较大可能对另一条路径感兴趣。因此，在路径推荐算法的实现中，需要考虑路径相似性的问题。

3 算法设计的基本思想

在本节中，将给出路径相似度算法与路径推荐功能的结合方式，以及设计路径相似度算法时的基本思想。

由于路径相似度算法是为应用中路径推荐功能所设计的，因此首先需要考虑与路径推荐功能的耦合方式，即在实现了路径相似度算法后如何利用路径相似度算法判定是否给用户推荐某条路径。设全体路径构成的集合为 $L = \{l_i \mid i \in \mathscr{A}\}$，定义集合 L 上的函数 $d : L \times L \to \mathbb{R}$，其中 $d(l_i, l_j)$ 表示路径 l_i 与路径 l_j 的相似度。并且，该函数 d 满足 $d(l_i, l_j) = d(l_j, l_i)$，$d(l_i, l_j) \in [0, 1]$，并定义常数 $\alpha, \beta \in [0, 1]$。若用户 A 创建或关注了路径 l_i，且对 $l_j \in L$ 有 $\alpha \leq d(l_i, l_j) \leq \beta$，则为用户 A 推荐路径 l_j，否则不推荐路径 l_j。上述耦合方式的基本想法为：用户一般会对与自己创建或关注的路径有共同点（如都包含某个电影，或包含其所创建或关注的路径中涉及电影的另一部电影中的地点，或包含与其创建或关注的路径中所涉及电影类型相似的电影的路径），但是又不完全相同的路径感兴趣。需要特别说明的是，不能仅因为相似

度较高就给用户推荐路径，因为这样可能造成推荐的路径与用户已经创建或关注的路径相同，或仅仅顺序不同，或其包含的地点是用户创建或关注的某一路径所包含的地点的子集，这显然不是用户希望被推荐的路线。

在设计路径相似算法时，首先需要考虑已经创建或关注了某些路径的用户会因为路径的哪些属性而对路径感兴趣。由于本应用是结合了电影/电视剧背景的旅游应用，因此除了传统旅游路径所涉及的因素外，还应当考虑电影/电视剧因素的影响。首先，就从传统的旅游路径推荐来讲，若用户计划在某个地区进行旅游，则其有较大可能对该地区的其他旅游路径感兴趣，即应考虑地点的区域属性。然后，从电影/电视剧的角度讲，若用户当前计划的路径中包含的地点对应的电影/电视剧在其他路径中也有出现，则用户也有较大可能对这些路径感兴趣，即应考虑路径中地点所对应的电影/电视剧与其他路径中包含的地点对应的电影/电视剧重复的属性。最后，还是从电影/电视剧的角度讲，若用户当前计划的路径中包含的地点对应的电影的另一部，或是姊妹篇，在其他路径所包含的地点中出现，则用户也有较大可能对这些路径感兴趣，即应考虑路径中地点所对应的电影/电视剧与其他路径中包含的地点对应的电影/电视剧的相似性。

另外，还需要考虑路径中哪些属性不会影响用户的兴趣。首先，与信号处理以及图像处理领域内的路径相似性不同，在旅游中，用户并不会关心路径具体的形状是否相同，不会因为两条路径的形状相似就会对这两条路径有相同的情感倾向。其次，由于本应用是以电影/电视剧为背景为用户推荐路线，因此用户真正感兴趣的是路线中所包含的地点及其背后所对应的电影/电视剧，因此路径中地点的先后顺序一般不会影响用户对一条路径的情感倾向。

综上所述，在设计路径的相似性算法时，需要考虑路径的区域属性，两条路径是否包含一定量对应相同电影/电视剧的地点，以及两条路径是否包含一定量对应相似电影/电视剧的地点这些因素。而相似性中不应体现的是路径的具体形状，以及路径中地点的先后顺序。

4 算法的形式化描述

在本节中，将给出路径相似度算法的形式化描述，包括输入的描述，算法计算过程的描述与解释，以及输出的描述。

设路径集为 $L = \{l_i \mid i \in \mathscr{L}\}$，地点集为 $P = \{p_i \mid i \in \mathscr{P}\}$，电影/电视剧集为 $M = \{m_i \mid i \in \mathscr{M}\}$，地区集为 $R = \{r_i \mid i \in \mathscr{R}\}$，地点到电影/电视剧集的映射为 $f : P \to 2^M$，电影/电视剧到与其相似的电影/电视剧的集合的映射为 $g : M \to 2^M$，地点到其所对应的区域的映射为 $h : P \to R$，并设路径 $l \in L$ 为地点的有穷序列 $l = p_1, p_2, \ldots, p_{n_l}$ $(n_l \in \{1, 2, \ldots\})$。其中，$L, P, M, D$ 皆为有穷非空集合，并且映射 f 满足：$\forall p \in P, f(p) \neq \emptyset$。

路径相似算法的输入为路径 l_i, l_j，其中 $l_i = p_1^{(i)}, p_2^{(i)}, \ldots, p_m^{(i)}$ $(p_k^{(i)} \in P), l_j = p_1^{(j)}, p_2^{(j)}, \ldots, p_n^{(j)}$ $(p_k^{(j)} \in P)$。

在路径相似度算法中，设有常数 $\alpha, \beta, \gamma \in [0, 1]$，并且 $\alpha + \beta + \gamma = 1$，$\alpha \geq \beta \geq \gamma$。本算法中将使用这三个常数来权衡根据路径的区域属性，两条路径是否包含一定量对应相同电影/电视剧的地点，以及两条路径是否包含一定量对应相似电影/电视剧的地点所得到的两条路线的相似度的分值，由此得到最终的分

值。我们记根据路径的区域属性得到的两条路线的相似度分值为 s_1，根据两条路径是否包含一定量对应相同电影/电视剧的地点所得到的分值为 s_2，根据两条路径是否包含一定量对应相似电影/电视剧的地点所得到的分值为 s_3，且有 $s_i \in [0,1]$ $(i=1,2,3)$，记将 s_1, s_2, s_3 重排列后得到的序列为 s'_1, s'_2, s'_3，则路径相似算法最终的输出 $d(l_i, l_j) = \alpha s'_1 + \beta s'_2 + \gamma s'_3$。即有 s_1, s_2, s_3 中最大者的权重最大，次大者权重次大，最小者权重最小。如此设计的原因为两个路径有较好的相似性仅仅需要满足路径所在区域大致相同，路径包含对应相同电影的地点，路径包含对应相似电影的地点这三个条件中的一者即可，但是两个路径完全相似，即 $d(l_i, l_j) = 1$，则要求完全符合这三个条件。通常，可取定 $\alpha = 0.7, \beta = 0.2, \gamma = 0.1$，而更细致的参数选择则可通过机器学习算法对数据进行分析得到。

下面将分别描述如何计算 s_1, s_2, s_3，即计算根据路径的区域属性，根据两条路径是否包含一定量对应相同电影/电视剧的地点，以及根据两条路径是否包含一定量对应相似电影/电视剧的地点所分别得到的两条路线的相似度的分值。

对于 s_1，即根据路径的区域属性计算路径 l_i, l_j 的相似度。有 $l_i = p_1^{(i)}, p_2^{(i)}, \ldots, p_m^{(i)}$ $(p_k^{(i)} \in P), l_j = p_1^{(j)}, p_2^{(j)}, \ldots, p_n^{(j)}$ $(p_k^{(j)} \in P)$，我们记这两条路径中所涉及到的全部区域构成的集合为 $R_{ij} = \{r \in R \mid \exists q \in \{1,2,\ldots,m\}, r = h(p_q^{(i)})$或者$\exists q \in \{1,2,\ldots,n\}, r = h(p_q^{(j)})\}$，并将 R_{ij} 中的区域排成序列 $r'_1, r'_2, \ldots, r'_{|R_{ij}|}$。然后，我们按如下方法对路径 l_i 定义向量 $z_i^{(1)}$：

$$z_i^{(1)} = \left(\frac{\left|\{p_k^{(i)} \mid h(p_k^{(i)}) = r'_1\}\right|}{m}, \frac{\left|\{p_k^{(i)} \mid h(p_k^{(i)}) = r'_2\}\right|}{m}, \ldots, \frac{\left|\{p_k^{(i)} \mid h(p_k^{(i)}) = r'_{|R_{ij}|}\}\right|}{m} \right)^T$$

同样，也按相似的方法对路径 l_j 定义向量 $z_j^{(1)}$：

$$z_j^{(1)} = \left(\frac{\left|\{p_k^{(j)} \mid h(p_k^{(j)}) = r'_1\}\right|}{n}, \frac{\left|\{p_k^{(j)} \mid h(p_k^{(j)}) = r'_2\}\right|}{n}, \ldots, \frac{\left|\{p_k^{(j)} \mid h(p_k^{(j)}) = r'_{|R_{ij}|}\}\right|}{n} \right)^T$$

在定义了向量 $z_i^{(1)}, z_j^{(1)}$ 后，我们定义离散型随机变量 $X_i^{(1)}, X_j^{(1)}$，其中 $X_i^{(1)}$ 的分布列为 $P(X_i^{(1)} = k) = z_i^{(1)}[k]$，$X_j^{(1)}$ 的分布列为 $P(X_j^{(1)} = k) = z_j^{(1)}[k]$，其中 $z_i^{(1)}[k]$ 表示向量 $z_i^{(1)}$ 的第 k 项，$z_j^{(1)}[k]$ 表示向量 $z_j^{(1)}$ 的第 k 项。显然，有 $P(X_i^{(1)} = k), P(X_j^{(1)} = k) \in [0,1]$ $(k = 1,2,\ldots,|R_{ij}|)$，且 $\sum_k P(X_i^{(1)} = k) = 1, \sum_k P(X_j^{(1)} = k) = 1$，满足分布列的定义。

因此，我们的问题可以转化为考虑随机变量 $X_i^{(1)}, X_j^{(1)}$ 之间分布的相似性，而两个分布之间的相似性，则可以通过将两个分布之间的距离通过一定转换得到。对于两个分布之间的距离，在此，我们采用 f-divergence 的一种特殊形式，即 total variation distance 来做为度量两个分布之间距离的函数，其为 f-divergence 中取 $f(t) = \frac{1}{2}|t-1|$ 时所得到的特殊形式。total variation distance 的定义如下：

定义 1 (Total Variation Distance). *设样本空间为 Ω，样本空间 Ω 上的 σ 代数为 $\mathcal{F}$。若有随机变量 X, Y，且随机变量 X 对应的概率测度为 P，随机变量 Y 对应的概率测度为 Q，则随机变量 X, Y 的 total variation distance $\delta(X,Y)$ 为：*

$$\delta(X,Y) = \sup_{A \in \mathcal{F}} |P(A) - Q(A)|$$

注意到我们只需要考虑离散型随机变量，因此在此给出在离散型随机变量下 total variation distance 的等价形式：

定理 1. *若随机变量 X, Y 为有限离散型随机变量，样本空间 Ω 为有限集，记随机变量 X 对应的概率测度为 P，随机变量 Y 对应的概率测度为 Q，则有：*

$$\delta(X,Y) = \sup_{A\in\mathcal{F}} |P(A) - Q(A)| = \frac{1}{2}\sum_{x\in\Omega} |P(x) - Q(x)|$$

由此，我们可以利用离散型随机变量下 total variation distance（下文简称为 TVD）的等价形式来计算随机变量 $X_i^{(1)}, X_j^{(1)}$ 之间的距离。另外，还注意到 TVD 必然满足 $\delta(X,Y) \in [0,1]$，因此我们可以定义 $s_1 = 1 - \delta(X_i^{(1)}, X_j^{(1)})$。这个定义满足 $s_1 \in [0,1]$，并且 s_1 随 l_i, l_j 的地区差异的增加而减小，符合根据路径的区域属性确定的相似度的要求。

对于 s_2，即根据两条路径是否包含一定量对应相同电影/电视剧的地点来计算路径 l_i, l_j 的相似度。同样的，有 $l_i = p_1^{(i)}, p_2^{(i)}, \ldots, p_m^{(i)}$ $(p_k^{(i)} \in P), l_j = p_1^{(j)}, p_2^{(j)}, \ldots, p_n^{(j)}$ $(p_k^{(j)} \in P)$，类似的，我们记这两条路径中所涉及到的全部电影/电视剧构成的集合为 $M_{ij} = \{m \in M \mid \exists q \in \{1,2,\ldots,m\}, m = f(p_q^{(i)})$或者$\exists q \in \{1,2,\ldots,n\}, m = f(p_q^{(j)})\}$，并将 M_{ij} 中的电影/电视剧排成序列 $m'_1, m'_2, \ldots, m'_{|M_{ij}|}$。然后，同样类似的，我们按如下方法对路径 l_i 定义向量 $z_i^{(2)}$：

$$z_i^{(2)} = \left(\frac{\left|\{p_k^{(i)} \mid m'_1 \in f(p_k^{(i)})\}\right|}{Z_i^{(2)}}, \frac{\left|\{p_k^{(i)} \mid m'_2 \in f(p_k^{(i)})\}\right|}{Z_i^{(2)}}, \ldots, \frac{\left|\{p_k^{(i)} \mid m'_{|M_{ij}|} \in f(p_k^{(i)})\}\right|}{Z_i^{(2)}} \right)^T$$

同样，也按类似的方法对路径 l_j 定义向量 $z_j^{(2)}$：

$$z_j^{(2)} = \left(\frac{\left|\{p_k^{(j)} \mid m'_1 \in f(p_k^{(j)})\}\right|}{Z_j^{(2)}}, \frac{\left|\{p_k^{(j)} \mid m'_2 \in f(p_k^{(j)})\}\right|}{Z_j^{(2)}}, \ldots, \frac{\left|\{p_k^{(j)} \mid m'_{|M_{ij}|} \in f(p_k^{(j)})\}\right|}{Z_j^{(2)}} \right)^T$$

其中，$Z_i^{(2)} = \sum_{l=1}^{|M_{ij}|} \left|\{p_k^{(i)} \mid m'_l \in f(p_k^{(i)})\}\right|, Z_j^{(2)} = \sum_{l=1}^{|M_{ij}|} \left|\{p_k^{(j)} \mid m'_l \in f(p_k^{(j)})\}\right|$。

在定义了向量 $z_i^{(2)}, z_j^{(2)}$ 后，我们可以同样定义离散型随机变量 $X_i^{(2)}, X_j^{(2)}$，其中 $X_i^{(2)}$ 的分布列为 $P(X_i^{(2)} = k) = z_i^{(2)}[k]$，$X_j^{(2)}$ 的分布列为 $P(X_j^{(2)} = k) = z_j^{(2)}[k]$，其中 $z_i^{(2)}[k]$ 表示向量 $z_i^{(2)}$ 的第 k 项，$z_j^{(2)}[k]$ 表示向量 $z_j^{(2)}$ 的第 k 项。显然，有 $P(X_i^{(2)} = k), P(X_j^{(2)} = k) \in [0,1]$ $(k = 1,2,\ldots,|M_{ij}|)$，且 $\sum_k P(X_i^{(2)} = k) = 1, \sum_k P(X_j^{(2)} = k) = 1$，满足分布列的定义。因此，类似的，我们可如计算 s_1 时一样利用 TVD 计算随机变量 $X_i^{(2)}, X_j^{(2)}$ 之间的距离 $\delta(X_i^{(2)}, X_j^{(2)})$，然后定义 $s_2 = 1 - \delta(X_i^{(2)}, X_j^{(2)})$。

最后，对于 s_3，即根据两条路径是否包含一定量对应相似电影/电视剧的地点来计算路径 l_i, l_j 的相似度。同样的，有 $l_i = p_1^{(i)}, p_2^{(i)}, \ldots, p_m^{(i)}$ $(p_k^{(i)} \in P), l_j = p_1^{(j)}, p_2^{(j)}, \ldots, p_n^{(j)}$ $(p_k^{(j)} \in P)$，进一步也有这两条路径中所涉及到的全部电影/电视剧构成的集合 M_{ij}。下面，我们首先将定义集合的映射闭包：

定义 2 (集合的映射闭包)**.** *若有非空有限集合 M，以及映射 $g : M \to 2^M$，称有限次使用下面两条规则构造的集合 $\widetilde{M}$ 为集合 M 在映射 g 下的映射闭包：*

1. $\forall m \in M, m \in \widetilde{M}$

2. $\forall m \in \widetilde{M}, g(m) \subseteq \widetilde{M}$

另外，也将 $\widetilde{M}$ 记为 $\mathrm{C}_g(M)$。

由映射闭包的定义，我们有集合 M_{ij} 在映射 g 下的映射闭包 $\widetilde{M}_{ij}$，该集合即表示这两条路径中所涉及到的与相关的全部电影/电视剧构成的集合。然后，我们将 $\widetilde{M}_{ij}$ 中的电影/电视剧排成序列 $m'_1, m'_2, \ldots, m'_{|\widetilde{M}_{ij}|}$。之后，同样类似的，我们按如下方法对路径 l_i 定义向量 $z_i^{(3)}$：

$$z_i^{(3)} = \left(\frac{\left|\{p_k^{(i)} \mid m'_1 \in \mathrm{C}_g(f(p_k^{(i)}))\}\right|}{Z_i^{(3)}}, \frac{\left|\{p_k^{(i)} \mid m'_2 \in \mathrm{C}_g(f(p_k^{(i)}))\}\right|}{Z_i^{(3)}}, \ldots, \frac{\left|\{p_k^{(i)} \mid m'_{|\widetilde{M}_{ij}|} \in \mathrm{C}_g(f(p_k^{(i)}))\}\right|}{Z_i^{(3)}} \right)^T$$

同样，也按类似的方法对路径 l_j 定义向量 $z_j^{(3)}$：

$$z_j^{(3)} = \left(\frac{\left|\{p_k^{(j)} \mid m'_1 \in \mathrm{C}_g(f(p_k^{(j)}))\}\right|}{Z_j^{(3)}}, \frac{\left|\{p_k^{(j)} \mid m'_2 \in \mathrm{C}_g(f(p_k^{(j)}))\}\right|}{Z_j^{(3)}}, \ldots, \frac{\left|\{p_k^{(j)} \mid m'_{|\widetilde{M}_{ij}|} \in \mathrm{C}_g(f(p_k^{(j)}))\}\right|}{Z_j^{(3)}} \right)^T$$

其中，$Z_i^{(3)} = \sum_{l=1}^{|\widetilde{M}_{ij}|} \left|\{p_k^{(i)} \mid m'_l \in \mathrm{C}_g(f(p_k^{(i)}))\}\right|, Z_j^{(3)} = \sum_{l=1}^{|\widetilde{M}_{ij}|} \left|\{p_k^{(j)} \mid m'_l \in \mathrm{C}_g(f(p_k^{(j)}))\}\right|$。

在定义了向量 $z_i^{(3)}, z_j^{(3)}$ 后，我们可以同样定义离散型随机变量 $X_i^{(3)}, X_j^{(3)}$，其中 $X_i^{(3)}$ 的分布列为 $P(X_i^{(3)} = k) = z_i^{(3)}[k]$，$X_j^{(3)}$ 的分布列为 $P(X_j^{(3)} = k) = z_j^{(3)}[k]$，其中 $z_i^{(3)}[k]$ 表示向量 $z_i^{(3)}$ 的第 k 项，$z_j^{(3)}[k]$ 表示向量 $z_j^{(3)}$ 的第 k 项。显然，有 $P(X_i^{(3)} = k), P(X_j^{(3)} = k) \in [0,1]$ $(k = 1, 2, \ldots, |\widetilde{M}_{ij}|)$，且 $\sum_k P(X_i^{(3)} = k) = 1, \sum_k P(X_j^{(3)} = k) = 1$，满足分布列的定义。因此，类似的，我们可如计算 s_1, s_2 时一样利用 TVD 计算随机变量 $X_i^{(3)}, X_j^{(3)}$ 之间的距离 $\delta(X_i^{(3)}, X_j^{(3)})$，然后定义 $s_3 = 1 - \delta(X_i^{(3)}, X_j^{(3)})$。

至此，在本节中，我们首先给出了对算法输入的描述，对算法输出的描述，以及对从 s_1, s_2, s_3 中计算出算法输出的过程的描述。然后，本节分别给出了计算 s_1, s_2, s_3 的方法。在计算 s_1, s_2, s_3 的过程中，本文首先将路径的相似性问题转换为概率分布的相似性问题，并进一步将概率分布的相似性问题转化为计算概率分布距离的问题，并最终使用 total variation distance 函数来解决度量概率分布之间距离的问题。特别的，在 s_3 的计算过程中，还给出了集合的映射闭包的定义，并借此得到了两条路径中所涉及到的与相关的全部电影/电视剧构成的集合，并进一步利用 TVD 计算出了 s_3。

5 改进与展望

本节中将探讨在未来对路径相似算法以及对路径推荐功能可能的改进方向。

就路径相似度算法的改进而言，由于本算法是基于与路径推荐功能相耦合的考虑，以及对用户可能感兴趣的路径进行分析所设计并进行参数选择的，设计时有较大的主观因素。若应用推出后有大量的用户参与，则可以考虑利用用户对所推荐的路径的点击统计与关注统计通过机器学习的手段来调整本算法中的相关参数，或者完全利用机器学习中如神经网络等手段来学习路径的相似度函数，由此利用实例来改进路径推荐的效率。

另外，在路径相似算法中本文所选择的刻画两个分布之间距离的 statistical distribution 为 total

variation distance，因此可考虑使用其他的 f-divergence，如 Kullback–Leibler divergence 或 Hellinger distance，或者其他常用的分布之间的距离，如 Wasserstein metric，Bhattacharyya distance，Lévy–Prokhorov metric，Jensen–Shannon divergence，以及 Rényi's divergence，并选择效果最好的距离函数来计算路径相似度。

就路径相似度算法所服务的路径推荐功能而言，在为用户推荐路径时，除了利用路径的相似度（即根据路径自身的属性进行推荐）外，还可考虑利用用户之间的关注关系所构成的关注图来为用户推荐路径，因为有关注关系的用户很可能对路径也有相似的喜好，相关的算法可以参考 Collaborative filtering algorithm。除了利用用户的关注关系之外，还可考虑全体用户的路径关注行为，挖掘“若用户 A 关注了路径 l_1，则其有多大可能关注路线 l_2”所对应的关联规则，利用数据挖掘中有关关联规则的算法给用户推荐路径，相关算法可参考 Apriori 算法，FG_growth 算法，H_mine 算法，OP 算法，以及 CLOSET+ 算法。最后，可考虑采用层次分析法在上述给出的基于路径自身的属性进行推荐，基于用户之间的关注关系进行推荐，以及基于路径之间的关联规则进行推荐的推荐结果中选出最终的路径进行推荐，并且层次分析法中的参数也可通过用户数据利用机器学习算法进行训练。

在路径推荐功能之外，为了能够达到更好的用户体验，还可考虑加入地点推荐功能，即在用户构建路径时直接为用户推荐相关地点。对于这个问题，可考虑在目前所创建的所有路径中执行关联规则的挖掘算法，挖掘地点之间的关联规则，来为用户在构建路径时推荐地点。