

实时路况制约下基于内容的兴趣点推荐

雷 凯¹, 刘树波¹, 李 丹², 李永凯¹

(1. 武汉大学 计算机学院, 武汉 430072; 2. 湖北省水利水电科学研究院, 武汉 430070)

摘 要: 传统兴趣点推荐方法缺乏对用户所处位置和实时路况信息的考虑, 推荐结果虽能满足用户偏好, 但会导致出行时间增加。为此, 提出一种考虑实时路况信息的推荐方法, 利用基于内容的方法获取兴趣点支持度, 使用实时路网状况获取实时路网支持度, 融合基于内容的支持度和实时路网支持度, 形成最终优化的推荐结果。实验结果表明, 与基于内容的推荐方法相比, 融合实时交通路况的推荐方法能节省更多的出行时间, 具有较强的实用性。

关键词: 兴趣点推荐; 基于内容的推荐; 实时路网; 用户偏好模型; 支持度

中文引用格式: 雷 凯, 刘树波, 李 丹, 等. 实时路况制约下基于内容的兴趣点推荐[J]. 计算机工程, 2017, 43(10): 147-152.

英文引用格式: LEI Kai, LIU Shubo, LI Dan, et al. Content-based POI Recommendation in Real-time Traffic Constraint[J]. Computer Engineering, 2017, 43(10): 147-152.

Content-based POI Recommendation in Real-time Traffic Constraint

LEI Kai¹, LIU Shubo¹, LI Dan², LI Yongkai¹

(1. School of Computer, Wuhan University, Wuhan 430072, China; 2. Hubei Water Resources Research Institute, Wuhan 430070, China)

[Abstract] Traditional Point of Interest (POI) recommendation method does not consider the user's location and real-time traffic information. Though it can satisfy the user's preference, it leads to a big increase in travel time. A recommended method of considering the real-time traffic information is proposed. It uses content-based recommendation method to get POI support, and uses the real-time traffic conditions to obtain real-time network support, fusion based on the POI support and real-time network support, and gets the final recommendation results. Experimental results show that, compared with the content-based recommendation method, the recommendation method can save more travel time and has strong practicability.

[Key words] Point of Interest (POI) recommendation; content-based recommendation; real-time traffic network; user preference model; support

DOI: 10.3969/j.issn.1000-3428.2017.10.025

0 概述

随着互联网+的发展,大量线下服务接入互联网,信息严重过载,使用户无从选择,因此,推荐系统应运而生。首先,推荐系统用于解决信息过载问题,帮助人们筛选出感兴趣的信息。其次,交通阻塞已经严重影响了人们的正常出行,若能在出行前得到最佳路径,既有利于提高办事效率,又能很大程度地缓解交通压力^[1]。针对此问题,在对用户推荐兴趣点时,如何既能满足用户偏好,又能考虑到用户当前地理位置和所处位置的交通状况,使得推荐结果更符合用户出行状况,值得深入研究。本文在互联网+交通背景下,提出一种考虑实时路况信息的推荐方法,该方法使用基于内容的推荐方法发现用户偏好,并考虑用户的当前位置和所处位置的实时交通路况,将2种情况对餐馆推荐的支持度进行融合,得出推荐结果。

1 相关研究

1.1 基于内容的推荐算法

个性化推荐系统于2005年由文献[2]提出。随着推荐技术研究的发展,许多学者提出了如协同过滤、基于内容、基于图^[3]等各种经典的推荐算法,并在此基础上不断改进创新。文献[4]使用基于内容的推荐方法,通过提取文本特征构造用户兴趣模型,将候选新闻与用户兴趣进行相似性计算得到推荐结果。为解决多样性问题,文献[5]用分区混合的推荐方法、协同过滤和基于内容方法各自产生推荐列表,按情况使得用户在不同列表选择新闻。文献[6]用网页上点赞、评论等行为进行反馈,判断用户在多样性和个性化方面的偏好产生附加权重,来提高候选新闻的推荐率,混合方法弥补了基于内容推荐多样性的不足,但却带来了冷启动问题。文献[7]提出将

基金项目: 国家重点基础研究发展计划项目“面向数字城市的实时跨媒体信息存储与公众服务”(2011CB302306)。

作者简介: 雷 凯(1994—),男,硕士研究生,主研方向为推荐系统;刘树波,教授、博士;李 丹、李永凯,博士研究生。

收稿日期: 2016-08-17 **修回日期:** 2016-10-21 **E-mail:** leikaikailk@163.com

基于内容的推荐方法和基于协同过滤的推荐方法融合,通过扩展关键字来提高推荐多样性。为进一步提高推荐适用性,文献[8]提出对上下文感知推荐系统的研究,文献[9]提出使用交通路况产生的物品间实际距离来替代传统的经纬度直线距离,使推荐结果更适用。而本文提出在实时路况下,结合实时路况信息和用户偏好来做推荐,使得推荐的结果更加适用于实际环境。

基于内容的推荐方法具体实现流程如下:

1) 构造待推荐项特征向量,提取活动信息中的关键字作为特征向量关键字,广泛采用向量空间模型 VSM^[10],计算分量权重常用方法有 TF-IDF^[11]。

2) 特征学习,根据偏好文档中用户偏好数据学习的用户偏好特征,常用算法有 KNN^[12] 最近邻方法、Rocchio^[13] 算法和决策树^[14] 算法。

3) 产生推荐,在前面两步基础上,比较候选项特征和用户偏好特征,将相关性较强的候选项推荐给用户。

1.2 基于城市交通网的最短路径问题

在实时路况中获得最优的路径,并结合最优路径和用户偏好来进行推荐的过程如下:首先需要得到最优路径,若将城市交通网看做网络图,最优路径问题就是网络图中节点之间最短路径问题。国内外研究者提出了多种最短路径求解方法,主要有 Dijkstra^[15], Floyd, A*^[16] 等。其中, Dijkstra 算法是目前已知理论上最完善的算法。

由于交通道路的突发性等特征,许多交通阻塞并非能够事先预知,因此考虑使用在线实时路况服务会更符合实际情况。

2 实时路况约束下的融合推荐模型及算法

本文在互联网+交通背景下,首先使用基于内容的方法与基于用户线上活动构造用户偏好模型(UIM)^[7],然后利用实时路况信息,构造实时路况模型(RTTM),最后融合 UIM 和 RTTM,构造考虑实时路况下的推荐模型 FUIT。首先对 FUIT 模型构建过程中的符号说明如下:

1) 主要特征词序列^[7]。对于兴趣点集合 $D = \{d_1, d_2, \dots, d_n\}$, 兴趣点 d_i 的主要特征词 f_k 来源于用户对其进行的评价,其内容为表征兴趣点内容和描述兴趣点特征的词汇或短语。令序列 $F = (f_1, f_2, \dots, f_k)$ 为 D 的主要特征词序列,这里 $f_i (1 \leq i \leq k)$ 为 D 的主要特征词, k 表示主要特征词的个数。

2) 用户偏好模型(UIM)^[7]。任意用户对其已评价的兴趣点根据其评论信息进行向量化处理并进行相应的数学运算,得到与 F 对应的权值向量构成用户偏好模型。其数学表达形式为 $UM = (wl_1, wl_2, \dots, wl_j, \dots, wl_k)$, 其中, wl_j 表示 F 中特征词 f_j 在

UIM 中的权值; UM 表示用户的偏好。

定义 1 实时路况模型(RTTM)。对于道路交通图 $G = (V, E)$, V 表示候选推荐兴趣点的集合, E 表示交通网络中主要道路集合。最短路径代价集合 $SP = (rts_1, rts_2, \dots, rts_j, \dots, rts_n)$, rts_j 代表到达第 j 个候选兴趣点的实时路况支持度。

定义 2 融合用户偏好和实时路况模型(FUIT)。对于任意用户,基于 UIM 推荐出的候选集支持度和考虑交通状况支持度融合,在融合的支持度下,从候选集中得到推荐结果,模型的构建流程及 FUIT 的算法流程如图 1 所示。

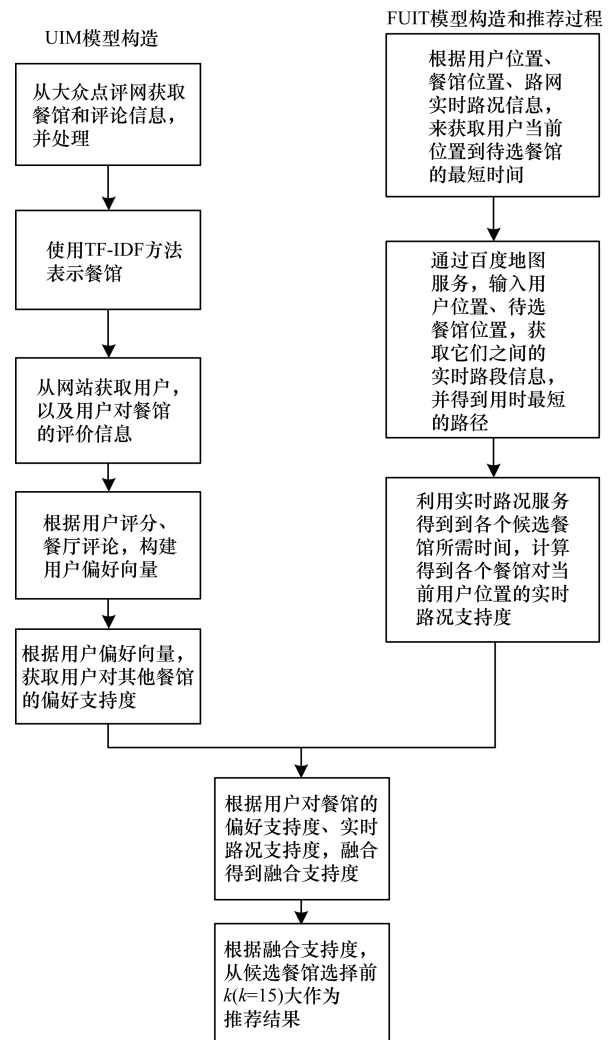


图 1 融合模型的构建和考虑实时路况下的算法流程

首先根据用户偏好构建出 UIM 模型, UIM 模型利用用户的偏好与餐馆相似度来进行推荐。然后考虑实时交通状况, 利用用户当前位置和餐馆位置, 在实时路况服务(百度地图)中获取从用户当前位置到达餐馆所需时间, 构建实时路况支持度, 融合餐馆的实时路况支持度与用户偏好支持度, 进而提出 FUIT。最后根据融合后的支持度, 生成更加合理的推荐结果。其中, 候选餐馆来自于所有用户预订过

的餐馆集合,路网匹配所需信息有用户实时位置、餐馆位置等信息。

2.1 用户偏好模型 UIM 的构建

在构建 UIM 时,兴趣点的特征提取自所有对其进行的评价,评价中出现多的短语或词语作为兴趣点特征。

给定兴趣点集合 $D = \{d_1, d_2, \dots, d_n\}$ 和主要特征词序列 $F = (f_1, f_2, \dots, f_k)$, 兴趣点 d_i 可以表示为与主要特征词序列 F 相对应的向量空间模型 (Vector Space Model, VSM) $d_i = (w_{i1}, w_{i2}, \dots, w_{ij}, \dots, w_{ik})$, 其中, w_{ij} 表示特征词 f_j 在兴趣点 d_i 评价中的权重值, $w_{ij} = 0$ 表示特征词 f_j 在兴趣点 d_i 的评论中没有出现过, 于是整个兴趣点集合可以表示为一个权值矩阵:

$$DM = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1k} \\ w_{21} & w_{22} & \cdots & w_{2k} \\ \vdots & \vdots & & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nk} \end{bmatrix} \quad (1)$$

本文使用 TF-IDF 表示法, 有的兴趣点评论多, 有的比较少, 为阻止评论多的兴趣点获得更高的权重, 使用式(2)计算权值 w_{ij} :

$$w_{ij} = [\text{freq}(i, j) / \maxOthers(i, j)] \times \log_a [N / n(j)] \quad (2)$$

其中, $\text{freq}(i, j)$ 为特征词 j 在兴趣点 i 评论中出现的次数; $\maxOthers(i, j)$ 为兴趣点 i 评论中其他特征词出现的最多次; N 为兴趣点总数; $n(j)$ 为出现过特征词 j 的兴趣点数目。

结合用户评论过的兴趣点和用户评分生成 UIM。首先, 定义用户评分对用户偏好的影响因子。

定义 3 评分影响因子 μ_i 。设用户 u 评论过的兴趣点集 $D_u = \{d_{u1}, d_{u2}, \dots, d_{um}\}$, 若其对兴趣点 d_{ui} 的评分为 p_{ui} , 则评分 p_{ui} 对用户 u 特征偏好的影响因子 μ_i 如式(3)所示。

$$\mu_i = \frac{p_{ui}}{\sum_{d_{uj} \in D_u} p_{uj}} \quad (3)$$

其中, 用户评论过的兴趣点集 D_u 是所有兴趣点集 D 的子集, 其权值矩阵 DM_u 为:

$$DM_u = \begin{bmatrix} w_{u11} & w_{u12} & \cdots & w_{u1k} \\ w_{u21} & w_{u22} & \cdots & w_{u2k} \\ \vdots & \vdots & & \vdots \\ w_{um1} & w_{um2} & \cdots & w_{umk} \end{bmatrix} \quad (4)$$

结合定义 3, 带评分影响的 UIM 推荐算法大致过程如下: 首先, 根据兴趣点的评论信息描述兴趣点的特征, 建立 DM 。然后, 根据用户参与过的兴趣点并结合评分, 计算用户偏好。最后, 根据用户偏好与待推荐兴趣点特征间的相似度进行推荐。

算法 1 考虑评分影响的基于偏好度的推荐算法

输入 用户 u 评论的兴趣点集合 D_u , 其权值矩阵 DM_u , 用户 u 对兴趣点 i 的评分 p_{ui} , 所有兴趣点集合 D , 推荐数目 n

输出 用户 u 的推荐兴趣点集合 S

Begin

Set $S = \emptyset$; Set 偏好向量 $UM = \{f_1, \dots, f_j, \dots\}, \forall j (f_j = 0)$;

Sum = 0;

For each 评论过的兴趣点 d_{ui} :

Sum + = p_{ui} ;

End for

For each 评论过的兴趣点 d_{ui} :

$U_i = p_{ui} / \text{Sum}$;

For each 特征值 $f_j \in$ 特征集合 F :

$f_j + = U_i \times w_{uij}$;

End for

End for

For each 兴趣点 $d \in D$:

兴趣点特征 d' 与 u 偏好的相似度 (余弦相似度)

$\text{sim} = \cos(UM, d')$;

If 相似度排序后的索引 $\text{index}(\text{sim}) < n$:

$S = S \cup \{d\}$;

End if

End for

End

2.2 考虑实时路况的推荐模型 FUIT

2.2.1 实时路况模型 RTTM 的建立

对给定用户推荐兴趣点时, 设用户的偏好向量为 $UM_u = (w_{u1}, w_{u2}, \dots, w_{uj}, \dots, w_{uk})$, 候选兴趣点特征向量为 $d = (wd_1, wd_2, \dots, wd_j, \dots, wd_k)$, 用余弦相似度求解用户偏好向量 UM_u 与兴趣点特征向量 d 的相似度。

候选兴趣点集合 $D' = (d'_1, d'_2, \dots)$, 设 d'_1 的位置为 (lon, lat) , 用户当前的位置为 (lon_u, lat_u) , 通过地图服务与实时路网进行匹配, 并结合实时交通状况, 构造路网交通图 $G = (V, E)$, 其中, V 为交通图中的节点, 对应道路的交叉口; E 为道路的路段。

从图 2 可以看出, 假定当前用户所在位置为路口 S , 若 UIM 推荐出的候选结果 d'_1 位于路口 D , d'_2 位于路口 F ; 假定 d'_1 推荐的支持度 $\text{sim}_{d'_1}$ 为 0.85, d'_2 的支持度 $\text{sim}_{d'_2}$ 为 0.84, 根据推荐结果来看, d'_1 优于 d'_2 , 但考虑实际路况, 从当前位置到候选点的最短所需时间分别为 $40 \text{ min} + 40 \text{ min} = 80 \text{ min}$, $10 \text{ min} + 10 \text{ min} = 20 \text{ min}$ 。因此, 若选择 d'_1 作为最佳推荐结果明显不符合实际情况, 在做推荐时需引入实时交通路况对推荐结果的影响, 根据用户的实际情况, 为用户选择相近偏好度但所需花费时间更少的兴趣点。

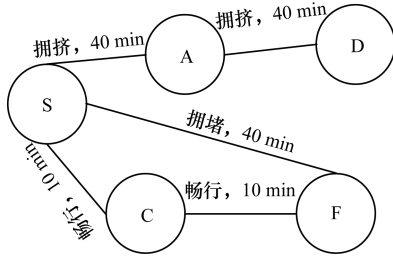


图 2 实时交通网络

定义 4 实时交通路况支持度。本文对于候选地点 d'_i 的实时交通路况支持度 $rts_{d'_i}$ 采用如下定义：

$$rts_{d'_i} = 1 - \frac{\min(\text{dis}(m, d'_i))}{\max_{d'_i \in D'}(\min(\text{dis}(m, d'_i)))} \quad (5)$$

其中, m 为用户当前位置; $\text{dis}(m, d'_i)$ 表示从当前位置 m 到候选地点 d'_i 路径; $\min(\text{dis}(m, d'_i))$ 表示从当前位置 m 到兴趣点 d'_i 的最短的路径所需时间; $\max_{d'_i \in D'}(\min(\text{dis}(m, d'_i)))$ 表示用户能够忍受的最大到达时间, 可以使用到所有候选项的最大到达时间或者自定义最大忍受时间。

定义 5 融合实时交通路况的支持度 (Fusion Traffic Support, $fts_{d'_i}$), 定义如下:

$$fts_{d'_i} = \alpha \text{sim}_{d'_i} + (1 - \alpha) rts_{d'_i}, 0 \leq \alpha \leq 1 \quad (6)$$

其中, $fts_{d'_i}$ 为融合实时交通路况后对 d'_i 的支持度; $\text{sim}_{d'_i}$ 为用户对 d'_i 的偏好支持度; $\alpha (0 \leq \alpha \leq 1)$ 为平衡用户偏好支持度和实时交通状况支持度的因子。

由上述可知, 在路网中, 考虑实时交通状况, 处于不同的位置的兴趣点对于用户而言, 要到达花费的时间不同, 基于使用户花费时间尽可能小的原则, 实时路况支持度越大的兴趣点, 使得用户花费时间就越小。实时路况兴趣点支持度计算方法如算法 2 所示。

算法 2 基于实时路况的兴趣点支持度算法 RealSup

输入 当前位置 m , 交通图 G , 兴趣点集 D'

输出 实时路况支持度集 SP

Begin

Set $SP = \emptyset$;

For each $d'_i \in D'$;

$$rts_{d'_i} = 1 - \frac{\min(\text{dis}(m, d'_i))}{\max_{d'_i \in D'}(\min(\text{dis}(m, d'_i)))};$$

$$SP = SP \cup \{rts_{d'_i}\};$$

End for

End

2.2.2 考虑实时交通路况的推荐算法

根据 2.1 节可知, 在实时路况下, 单纯考虑用户偏好的推荐系统并不合理, 应充分考虑用户位置和实时路况信息才更合理。算法大致过程如下: 首先, 根据用户在系统中的行为, 考虑用户的偏好, 并得出用户对不同兴趣点的偏好支持度。其次, 根据用户

所处实际位置, 结合路网实时交通路况, 计算各兴趣点对用户实时位置的支持度。最后, 结合用户偏好支持度和实时路况支持度, 得出融合支持度, 最终取前 n 个作为推荐结果。

算法 3 融合实时交通路况的推荐算法

输入 用户 u 当前位置 m , 评论过的兴趣点集合 D_u , 其权值矩阵 DM_u , 用户 u 对兴趣点 i 的评分 p_{ui} , 所有兴趣点集合 D , 交通图 G , 推荐数目 n

输出 用户 u 的推荐兴趣点集合 S

Begin

Set $S = \emptyset$; 偏好向量 $UM = \{f_1, f_2, \dots, f_j, \dots\}, \forall j(f_j = 0)$;

Sum = 0;

For each 评论过的兴趣点 d_{ui} :

Sum + = p_{ui} ;

End for

For each 评论过的兴趣点 d_{ui} :

$U_i = p_{ui} / \text{sum}$;

For each 特征值 $f_j \in$ 特征集合 F :

$f_j + = U_i \times W_{uij}$;

End for

End for

$SP = \text{RealSup}(m, G, D)$;

For each 兴趣点 $d \in D$:

兴趣点特征 d' 与 u 偏好的相似度 (余弦相似度) $\text{sim}_d = \cos(UM, d')$;

兴趣点 d 对用户的实时路况支持度 $SP_d = \text{sp}[d]$;

融合的支持度 $fts_d = \alpha \text{sim}_d + (1 - \alpha) SP_d$;

If 融合支持度排序后索引 $\text{index}(fts_d) < n$:

$S = S \cup \{d\}$;

End if

End for

End

3 实验及验证

3.1 评价指标

本文使用平均所需时间 C_a 作为实验的评价指标, C_a 定义如式 (7) 所示。

$$C_a = (\sum_{t \in T} t) / n \quad (7)$$

其中, $t = \min(\text{dis}(m, d'_i))$ 表示当前位置 m 到兴趣点 d'_i 所需最短路径时间; T 表示到推荐兴趣点集合所需最短路径时间的集合; n 表示推荐兴趣点集大小, 若推荐结果在实时路况下所需时间越短, 则 C_a 越小。

融合偏好度的平均代价 C_{FI} , C_{FI} 定义如式 (8) 所示。

$$C_{FI} = \left(\sum_{t \in T} \left(\alpha \frac{t}{t_{\max}} + (1 - \alpha) (1 - \text{sim}_{d'_i}) \right) \right) / n \quad (8)$$

其中, $\alpha (0 \leq \alpha \leq 1)$ 代表偏好度和实时路况的融合程度; 令 $t_{\max} = \max\{t | t \in T\}$, 或令 t_{\max} 为自定义最长可容忍超时时间, 表示所选路径的时间阈值, 若超过此时间则从推荐列表剔除; n 表示推荐兴趣点集大小;

sim_{d_i} 表示用户偏好支持度。由式(8)可知,花费时间越少,用户偏好度越高,则 C_{FI} 越小。

推荐结果的准确率 P , P 定义如式(9)所示。

$$P = \text{推荐且被接受的餐馆} / \text{推荐餐馆} \quad (9)$$

推荐结果的召回率 R , R 定义如式(10)所示。

$$R = \text{推荐且被接受的餐馆} / \text{被接受的餐馆} \quad (10)$$

3.2 实验方案

本文以餐馆推荐为例,使用 python 工具库 requests, bs4 获取网络数据,使用 pandas 进行数据处理,从大众点评网站获取的在武昌区与洪山区包含的4个商业区徐东大街、中南路、街道口、光谷,共计1248家店、278011条评论,从每一条评论中抽取简略描述短语作为关键字。实验使用多个用户的点评信息模拟对用户推荐。首先使用第2.1节提出的 UIM 方案获取推荐结果。然后结合当前的路网状况,使用百度地图路况 API 获取用户当前实时位置到各个商业区的最短所需时间和其经过的路径,再使用第2.2节提出的 FUIT 方案获取融合后的推荐结果,最后对比2种不同方案获取结果的平均所需时间代价 C_a 和融合偏好度的平均所需时间代价 C_{FI} 。

每个店家的评论抽取的关键字如图3所示。

id	good	bad
0,24037716	回头客(21);份量足(11);闺蜜聚会(8);朋友聚餐(6);约会圣地(4);下午茶(3);点心	1,66097614;点心不错(3);主食不错(3);米线不错(3);口味一般(3);服务一般(3)
2,15854806	回头客(62);有表演(23);朋友聚餐(16);请客(10);约会圣地(8);高端大气(7);闺蜜	3,16814053;回头客(163);下午茶(47);朋友聚餐(42);份量足(35);约会圣地(35);闺蜜聚会(20)
4,16961894	回头客(100);朋友聚餐(23);约会圣地(16);份量足(12);闺蜜聚会(10);家庭聚餐(4	5,19424508;回头客(47);份量足(20);请客(20);朋友聚餐(19);下午茶(11);闺蜜聚会(9);影

图3 关键字列表

其中,店家的特征就是由诸如回头客、朋友聚餐、肉类不错等用户评价中抽取出来的信息构成,做出此评价的用户数目就是此项特征的频率。

3.3 实验结果及分析

首先根据 UIM 方案,根据用户偏好获取对用户的推荐列表,设推荐餐馆集大小 n 为15。

然后使用 FUIT 方案,令 α 取值0.5,设最大可容忍时间为60 min,通过百度路况 API 获取18:00—19:10时间段内,丁字桥和武泰闸附近的实时路况信息,其到两大商业区光谷和徐东大街的最短时间如图4所示。

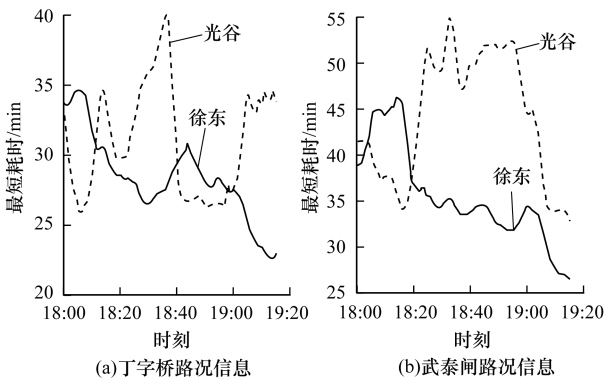


图4 实时路况时间

根据所处不同位置,实时路网状况不同,推荐的结果也不同,对比 UIM 方案、FUIT 方案的推荐结果,使用平均所需时间 C_a 作为评价指标,对多个用户进行推荐求得 C_a ,并取平均,所得结果如图5所示。

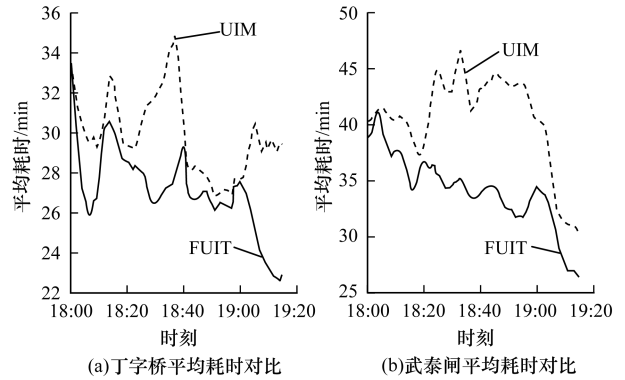


图5 推荐结果平均耗时对比

使用融合偏好度的平均代价 C_{FI} 作为评价指标,对多个用户进行推荐获取 C_{FI} ,并取平均,对比 UIM 方案、FUIT 方案的推荐结果,所得结果如图6所示。

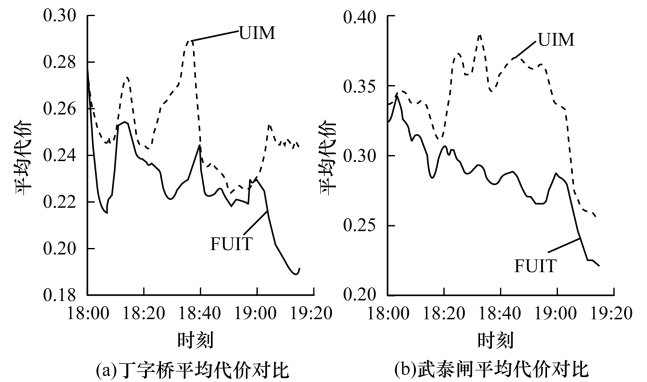


图6 推荐结果平均代价对比

从图6可以看出,考虑实时交通状况的推荐结果比原始的推荐结果更符合用户的实际出行状况。

为了进一步验证推荐的准确率和召回率等特征,使用半模拟数据集^[8]来验证,实验通过对实验室多名成员进行在线测试的形式,并记录在不考虑路况时,每个人所意愿选择的餐厅。然后将所有餐厅合并在一起生成餐厅集 G ,再针对每个人考虑实时路况,令其从 G 中选择其意愿选择餐厅,得出在考虑实时路况情况下,人们在选择餐馆时的一种规则,即80%的人们倾向于选择相对耗时少的商业区的餐馆。然后将此种规则应用于原始的未考虑实时路况情况下的用户订餐数据集,生成新的数据集。新的数据集包含了用户当前位置、所选择餐厅及所处交通状况,即当前位置到达各个商业区的实时时间。最后对生成的新的数据集进行分析,使用随机80%作为训练数据,20%作为测试数据。

分析 FUIT 方案与 UIM 方案的准确率指标,如图7所示。

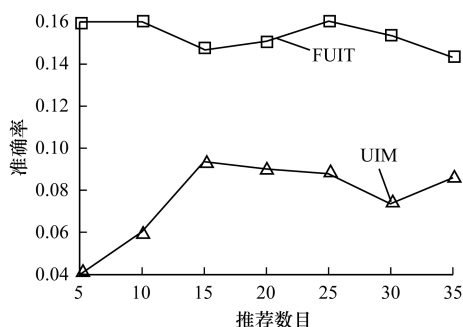


图7 推荐结果准确率对比

分析 FUIT 方案与 UIM 方案的召回率指标,如图 8 所示。

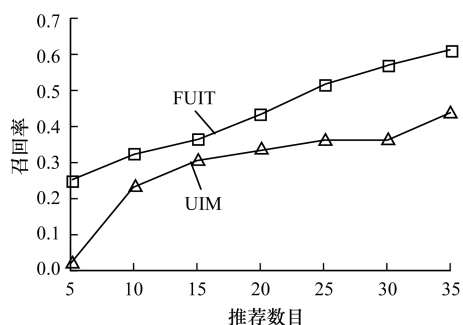


图8 推荐结果召回率对比

从图 8 可以发现,使用基于内容推荐方法的 UIM 方案在推荐准确率上效果不是很好,考虑到实时交通路况信息后,在同等条件下,基于人们会优先考虑交通方便的出行地点这一准则,考虑到交通路况信息的 FUIT 方案在准确率和召回率等指标上会更优,在推荐数目约为 15 时推荐效果比较好。

4 结束语

本文在互联网+交通背景下,提出一种实时路况制约下基于内容的餐馆推荐方法,该方法使用基于内容的推荐方法发现用户偏好,并考虑用户的当前位置和所处位置的实时交通路况,将 2 种情况对餐馆推荐的支持度进行融合,得出推荐结果。在结合实时路况情况下,考虑用户当前位置到餐馆的最佳路径,然后通过实验验证。融合实时路况后的推荐结果考虑了用户的既有偏好和实时交通状况,更能满足用户的需求。下一步将考虑从传统的资源推荐向融合了社会关系的社会推荐过渡,进一步提高推荐系统在实际应用中的适用性。

参考文献

- [1] 龙其,叶晨,张亚英.动态路网中基于实时路况信息的分布式路径生成算法[J].计算机科学,2014,41(9):259-262.
- [2] ADOMAVICIUS G, TUZHILIN A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions [J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6):734-749.
- [3] 马宏伟,张光卫,李鹏.协同过滤推荐算法综述[J].小型微型计算机系统,2009,30(7):1282-1288.
- [4] LI L, CHU W, LANGFORD J, et al. A Contextual-bandit Approach to Personalized News Article Recommendation [C]//Proceedings of the 19th International Conference on World Wide Web. New York, USA: ACM Press, 2010:661-670.
- [5] WASFI A M A. Collecting User Access Patterns for Building User Profiles and Collaborative Filtering [C]//Proceedings of the 4th International Conference on Intelligent User Interfaces. New York, USA: ACM Press, 1998:57-64.
- [6] WEN H, FANG L, GUAN L. A Hybrid Approach for Personalized Recommendation of News on the Web [J]. Expert Systems with Applications, 2012, 39(5):5806-5814.
- [7] 杨武,唐瑞,卢玲.基于内容的推荐与协同过滤融合的新闻推荐方法[J].计算机应用,2016,36(2):414-418.
- [8] 王立才,孟祥武,张玉洁.上下文感知推荐系统[J].软件学报,2012,23(1):1-20.
- [9] 邵阔义,班晓娟,王笑琨,等.基于交通网络数据优化的地理信息推荐系统[J].工程科学学报,2015,37(12):1651-1657.
- [10] 张振峰.基于向量空间模型的文本分类算法研究[D].杭州:杭州电子科技大学,2011.
- [11] 李镇君,周竹荣.基于 Document Triage 的 TF-IDF 算法的改进[J].计算机应用,2015,35(12):3506-3510.
- [12] 罗贤锋,祝胜林,陈泽健,等.基于 K-Medoids 聚类的改进 KNN 文本分类算法[J].计算机工程与设计,2014,35(11):3864-3867.
- [13] 郑秋生,翟琳琳.基于改进 Rocchio 算法的短文本自动分类研究[J].中原工学院学报,2013,24(1):70-73.
- [14] 邓一贵,伍玉英.基于文本内容的敏感词决策树信息过滤算法[J].计算机工程,2014,40(9):300-304.
- [15] 王树西,李安渝. Dijkstra 算法中的多邻接点与多条最短路径问题[J].计算机科学,2014,41(6):217-224.
- [16] 张翼,唐国金,陈磊.时相关车辆路径规划问题的改进 A* 算法[J].控制工程,2012,19(5):750-762.

编辑 索书志