文章编号：2095-2163(2023)06-0030-09 中图分类号：TP311 文献标志码：A

基于时间的空间文本关键词 skyline 查询

李晨阳[1]，董雷刚[2]，孙国豪[3]，于 泉[4]

（1 吉林化工学院 信息与控制工程学院，吉林 吉林 132022；2 白城师范学院 计算机科学学院，吉林 白城 137000；
3 东华大学 计算机科学与技术学院，上海 201620；4 蚂蚁科技集团股份有限公司，杭州 310012）

摘 要：在移动互联网环境下，空间文本 skyline 查询可以有效支持用户在空间和关键词方面的查询。随着需求的多样性，基于用户经常会同时考虑空间距离、数值型信息、关键词和时间等因素对查询结果的影响，提出了基于时间的空间文本关键词 skyline 查询（Time based Spatial Text Keyword Skyline Query，TSTKSQ），用来查找在空间、数值、关键词和时间都满足条件的优秀对象，设计了基于时间的空间文本关键词 skyline 查询的索引结构 STTR-Tree，提出了关键词、时间和时空关键词相关性的评价函数，在裁剪策略的基础上提出了 skyline 查询算法。通过实验结果分析，验证了算法的准确性和有效性。

关键词：空间文本 skyline 查询；关键词相关性；时间相关性；时空关键词相关性；STTR-Tree 索引

Time based spatial text keyword skyline query

LI Chenyang[1], DONG Leigang[2], SUN Guohao[3], YU Quan[4]

（1 School of Information and Control Engineering, Jilin Institute of Chemical Technology, Jilin 132022, China;
2 School of Computer Science, Baicheng Normal University, Baicheng Jilin 137000, China;
3 School of Computer Science and Technology, Donghua University, Shanghai 201620, China;
4 Ant Technology Group Co., Ltd, Hangzhou 310012, China）

【Abstract】 In the mobile Internet environment, spatial text skyline queries can effectively support users′ queries in terms of space and keywords. With the diversity of needs, based on the fact that users often consider the influence of spatial distance, numerical information, keywords and time on query results at the same time, a Time based Spatial Text Keyword Skyline Query (TSTKSQ) is proposed to find the spatial, numerical, keyword and time are satisfied with the conditions of the excellent object. The index structure STTR-Tree for time based spatial text keyword skyline query is designed, the evaluation functions of keyword, time and spatio-temporal keyword relevance is proposed, and the skyline query algorithm is proposed on the basis of the tailoring strategy. The accuracy and effectiveness of the algorithm are verified through the analysis of experimental results.

【Key words】 spatial text skyline query; keywords relevance; time correlation; temporal spatial keyword relevance; STTR-Tree index

0 引 言

科技的发展产生了海量的数据信息，在移动通信和互联网技术快速发展的背景下，用户对互联网中的数据信息提出了具有特定的查询需求。2001 年，Börzsönyi 等人在文献[1]中首次将 skyline 查询应用于数据库领域，作为一种高效的数据检索方式，被广泛应用于多目标决策、市场分析和数据挖掘等多个领域中。Skyline 查询的结果为一组 skyline 对象，这些 skyline 对象均不能被同一数据集中其它任何对象支配。

在实际应用中，用户对查询的要求越来越多，现有的空间文本 skyline 查询算法在计算时考虑的因素较少，无法满足用户需求。例如，用户计划在某天晚上与朋友聚餐，需要预定一个 20:00-22:00 时间段可以营业、距离火车站近、价格低、服务质量好，且

基金项目：吉林省自然科学基金项目（YDZJ202201ZYTS666）；吉林省教育厅科学研究项目（JJKH20210005KJ）。

作者简介：李晨阳（1999-），男，硕士研究生，主要研究方向：数据查询与优化；董雷刚（1982-），男，博士，副教授，硕士生导师，主要研究方向：数据查询与优化；孙国豪（1990-），男，博士，副教授，主要研究方向：大数据；于 泉（1991-），男，学士，高级工程师，主要研究方向：数据挖掘。

通讯作者：董雷刚 Email：Lgdong010@163.com

收稿日期：2023-02-21

最好拥有停车场的饭店。在表 1 中列出了 4 个饭店的信息，包含了饭店到查询点的空间距离、饭店的人均消费价格、用户评分、关键词信息以及营业时间。

表 1　饭店信息

Tab. 1　Hotel information

饭店名称	空间距离/Km	人均价格/元	用户评分	关键词	营业时间
a	3.6	90	8	停车场、空调	5:30-9:00
b	4	60	7	wifi、空调	10:00-22:00
c	2.2	90	8	停车场	22:00-3:00
d	4.5	80	7	wifi、停车场	11:00-14:00、17:30-24:00

由于此类查询同时包括空间位置、数值型信息、关键词以及时间 4 个属性，以往的空间文本 skyline 查询不能直接解决此类问题。如，文献[2]中提出了空间多关键词 skyline 查询算法 SKS，将空间距离和文本相似度相结合，建立了加权距离的空间文本支配模型。SKS 算法主要考虑了加权距离和数值型属性，并没有考虑时间属性对查询结果的影响。文献[3]中提出了已知时间的空间文本 skyline 查询 TSTSQ，TSTSQ 中考虑了查询点和对象间的空间距离、查询关键词与对象包含的关键词间的文本相关性以及查询时间段和对象包含时间段的时间相关性 3 个属性。在查询时通过计算空间文本相关性函数 $kd(q,o)$ 和时间文本相关性函数 $kt(q,o)$ 来判断对象间的支配关系。然而，此查询并没有考虑数值型信息对查询结果的影响，查询结果集具有一定的缺陷。当前文献考虑的都是时间、空间距离、数值型信息和关键词中的若干个因素，并没有将这 4 个因素同时考虑进去进行研究，而同时考虑这 4 个因素后将会为用户返回更适合用户偏好的结果集。

基于此，本文将时间、空间距离、数值型信息和关键词几个因素相结合，提出一种基于时间的空间文本关键词 skyline 查询，构建基于时间的空间文本关键词 skyline 查询的索引结构以及查询算法，满足用户更多和更具体的查询需求。

1　相关工作

最开始对 skyline 查询的研究是以数值型属性为支配判断条件找到最优候选集，文献[4]中介绍了最近邻 NN 算法和分支界限 BBS 算法，其中，最近邻搜索策略是基于 R^*-Tree 索引对象，BBS 算法是在 NN 算法的基础上进行改进，BBS 算法只对可能包含 skyline 点的 R 树节点进行访问，不会重复检索，其内存开销明显小于 NN 算法。然而，上述查询没有考虑空间属性对查询结果的影响。随着进一步的研究，文献[5]考虑了空间属性，提出了欧式空间和路网空间中的 skyline 查询问题；文献[6]中将 K-支配应用到道路网 skyline 查询中，提出了道路网环境下 K-支配空间 skyline 查询方法，来处理多属性数据对象。在实际应用中只考虑空间属性并不能满足用户的偏好性需求，用户的偏好性需求一般通过关键词等文本信息来描述，文献[7]提出将空间位置和查询关键词作为查询条件，使用 Voronoi 进行空间数据管理，建立路网中每个点的主导区域来求解最优查询结果。考虑在实际应用中欧式距离的局限性，文献[8]提出了基于曼哈顿距离的空间 skyline 查询；文献[9]提出使用 R^* 树索引空间和文本数据，文本数据采用倒排文件索引结构，并添加到 R^* 树上，该索引结构插入数据的速度比 R 树快，并且比传统的空间索引花费时间少；文献[10]提出了加权空间 skyline 查询，每个兴趣点都有不同的重要性，给每个兴趣点分配不同的权重，并使用加权欧几里得距离来获取 skyline 点集。

以上文献虽然在一定程度上解决了 skyline 查询和空间文本 skyline 查询等问题，但随着用户的偏好性需求不断增加，以往的 skyline 查询已经不能满足用户的需求，需要考虑其他因素对查询结果的影响。在移动互联网环境下，文献[11]将方向这一属性应用到空间 skyline 查询中，提出了基于方向的空间 skyline，该查询从不同方向检索最优候选对象，查询结果为不同方向上的 skyline 对象，并提出了伪 skyline 的概念，如果某一方向上没有 skyline 对象，则用伪 skyline 对象替代。考虑到用户社交对查询的影响，文献[12]提出了基于社交的空间文本 skyline 查询，设计了新的评价函数来计算用户的社交相关性。为了提高查询结果的质量，引入了受限 skyline，当 skyline 查询返回的结果少于设定的阈值时，需要进行受限 skyline 查找，最后返回的是 skyline 对象和受限 skyline 对象。文献[13]将空间关键字查询与社交数据相结合，提出了路网地理社交 top-k 和 skyline 关键词查询，通过对象的空间信

息、文本信息和社交网络信息来进行查询。考虑到时间在查询中的重要作用。文献[14]将时间信息与空间关键词查询结合,同时考虑对象与查询点之间的位置相关性、文本相关性和时间相关性,并且定义了两个评价函数来满足用户的不同需求。文献[15]提出了在路网中有效处理具有时变属性的对象的 skyline 查询问题。文献[16]将时间属性应用到 Top-k 查询中,根据用户的空间位置和时间,为用户返回 k 条旅行时间最短的路线。

综上所述,现有算法并不能解决带有时间的空间文本关键词 skyline 查询问题,因此本文提出一种基于时间的空间文本关键词 skyline 查询,获得那些在时间、空间、文本、数值 4 个方面具有最优表现的对象集合,以满足用户更具体的偏好需求。

2 问题定义

为了清晰地判断对象间的支配关系,本节将着重介绍查询点 q 与对象 o 之间的空间距离、关键词相关性、时间相关性以及时空关键词相关性的评价函数。

2.1 空间距离

$$SD(q,o) = d(q,o) \tag{1}$$

其中,$d(q,o)$ 表示查询点和对象点间的欧式距离,则查询点与对象点间的空间距离就是两点间的欧式距离。

2.2 关键词相关性

假设查询关键词有 n 个,对象包含的关键词有 m 个,则有

$$\mathrm{KR}(q,o) = \sum_{i=1}^{n} V_i \tag{2}$$

$$V_i = \begin{cases} \omega_i & |qk_i \cap ok_j| \neq 0 \\ 0 & |qk_i \cap ok_j| = 0 \end{cases} (i \in [1,n], j \in [1,m]) \tag{3}$$

其中,$|qk_i \cap ok_j| \neq 0$ 表示查询关键词与对象包含的关键词相交;$|qk_i \cap ok_j| = 0$ 表示查询关键词与对象包含的关键词不相交;ω_i 表示查询关键词的权重。

每个查询关键词的权重有两种设定情况,一是由用户根据偏好对每个查询关键词进行设定,其二是默认所有查询关键词的权重相等。V_i 表示每个查询关键词与对象包含的关键词的相关性,则关键词相关性就是每个查询关键词与对象包含的关键词的相关性之和。

以表 1 中包含的对象为例,其中包含了饭店到查询点的空间距离、饭店的人均消费价格、用户评分、关键词信息以及营业时间等信息。假设用户查询的关键词为“wifi”和“空调”,关键词的权重根据用户的偏好设定,设用户对“wifi”的偏好权重为0.6,对“空调”的偏好权重为 0.4,则对象 a、b、c、d 的关键词相关性分别为 0.4、1、0、0.6,如果用户没有设置关键词的权重,则默认所有查询关键词的权重相等,此时对象 a、b、c、d 的关键词相关性分别为 0.5、1、0、0.5。

2.3 时间相关性

$$TC(q,o) = \frac{|qtq \cap otq|}{|qtq|} \tag{4}$$

其中,$|qtq \cap otq|$ 表示查询时间段与对象包含的时间段之间相交的数值,$|qtq|$ 表示查询时间段的数值,则时间相关性就是查询时间段和对象包含的时间段间相交的数值与查询时间段的数值的比值。以表 1 中包含的对象为例,假设用户查询的时间段是 20:00 - 22:00,则对象 a、b、c、d 的时间相关性分别为 0、1、0、1。

为了对某个对象的空间距离、关键词相关性及时间相关性有一个综合评价,本文提出了时空关键词相关性函数来衡量一个对象同时在空间、时间、文本上的优劣程度。其中,α 是一个平衡系数,用来平衡关键词相关性与时间相关性间的权重,在没有用户设定的情况下,默认二者权重相等。本文设定时空关键词相关性的数值越小对象越优。

2.4 时空关键词相关性

$$\mathrm{TSKR}(q,o) = \frac{SD(q,o)}{\alpha \mathrm{KR}(q,o) + (1-\alpha)\mathrm{TC}(q,o)} \tag{5}$$

以表 1 中包含的对象为例,假设用户查询的关键词为“wifi”和“空调”,用户查询的时间段是 20:00-22:00,默认所有查询关键词的权重相等。根据计算,对象 c 的关键词相关性为 0,对象 a 和对象 c 的时间相关性都为 0。因此,对象 a、c 不必进行计算可以根据算法提前裁剪,而对象 b、d 的时空关键词相关性分别为 4、6,说明对象 b 优于 d。

定义 1(数值型信息支配) 给定数据集中具有 n 维数值型属性的任意两个对象 o_i、o_j,如果 o_i 在其 m 维数值型属性中至少有一维属性优于 o_j,则称在 m 维数值型属性上 o_i 支配 o_j,记为 $o_i \prec_M o_j$。

本文设定数值型属性的数值越小对象表现越优,但在表 1 中用户评分属性一般是数值越大越好。如果遇到某一数值型属性值越大对象越优的情况,则先将对象 o 进行预处理:$o_i' = \max_i - o_i$,其中 $\max_i$ 表示第 i 维数值型属性的最大值,o_i 表示对象 o 在第 i 维数值型属性的取值。

定义 2（基于时间的空间文本关键词支配）给定查询点 q 和空间数据集 D 中的任意两个对象 o_i、o_j，如果 o_i、o_j 同时满足 $o_i \prec_{NI} o_j$ 且 TSKR(q,o_i) ≤ TSKR(q,o_j)，则称 o_i 基于时间的空间文本关键词支配 o_j，记为 $o_i \prec_{TSTK} o_j$。

以表 1 中包含的对象为例，假设用户需要预定晚上 20:00-22:00 与其当前位置距离近、价格低、服务质量好，最好拥有“wifi”和“空调”的饭店。根据计算对象 b、d 的时空关键词相关性分别为 4、6，并且根据对象 b 和 d 的数值型信息可知 $b \prec_{NI} d$，所以由定义 2 可知，$b \prec_{TSTK} d$。

定义 3（基于时间的空间文本关键词 skyline）给定一个数据集 D，基于时间的空间文本关键词 skyline 就是从该数据集中返回那些不能被其它任何对象支配对象的集合。即，当且仅当 $\forall o' \in D$、$o' \not\prec_{TSTK} o$ 时 $o \in$ TSTKS。

由定义 2 中的例子可得，基于时间的空间文本关键词 skyline 为 $\{b\}$。

3　STTR-Tree 索引

为了高效地获取 skyline 对象，需要建立相关索引结构。虽然 R-Tree[17] 是一种经典的空间索引数据结构，但其只包含对象的空间信息，随后学者们又提出了 IR-Tree[18]、IR^2-Tree[19] 等空间索引，也不能同时存储对象的空间、数值型信息、关键词及时间等信息。因此，本文提出一种可以同时存储对象的空间、数值型信息、关键词及时间等信息的 STTR-Tree 索引。STTR-Tree 索引结构如图 1 所示。

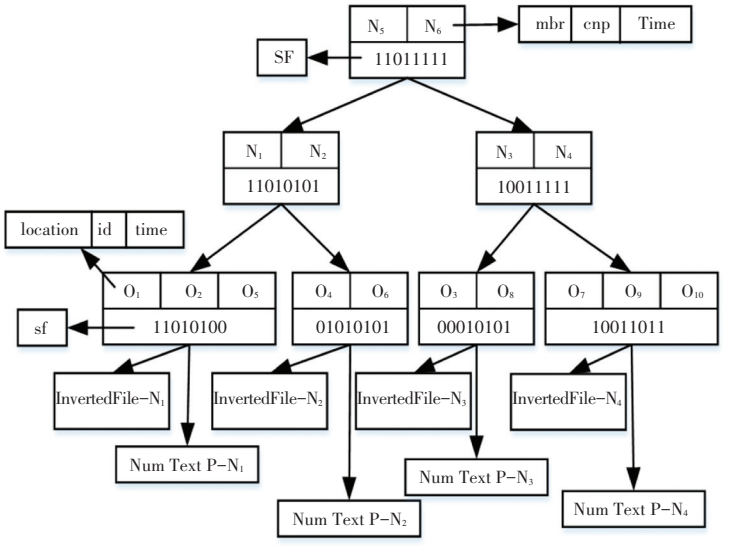

图 1　STTR-Tree 索引

Fig. 1　STTR-Tree index

STTR-Tree 索引中叶子结点主要包含以下信息：对象的空间位置信息（location）、对象在数据集中的标识符（id）、对象包含的时间段信息（time）、指向该结点的文件倒排表的指针（InvertedFile）。文件倒排表中的关键词是由该结点包含的所有对象关键词的并集组成。对象 o_1、o_2、o_4、o_5、o_6 包含的时间段信息见表 2，叶子结点的文件倒排表见表 3。

表 2　对象的时间段信息

Tab. 2　Time period information of objects

对象	时间段
O_1	6:00-8:00
O_2	9:00-12:00
O_4	10:00-17:00
O_5	8:00-20:00
O_6	18:00-24:00

表 3　叶子结点文件倒排表

Tab. 3　Inverted list of leaf node files

InvertedFile-N_1	InvertedFile-N_2	InvertedFile-N_3	InvertedFile-N_4
k_1:o_1、o_2	k_1:	k_1:	k_1:o_9、o_{10}
k_2:o_5	k_2:o_6	k_2:	k_2:
k_3:	k_3:	k_3:	k_3:
k_4:o_2、o_5	k_4:o_4	k_4:o_3、o_8	k_4:o_7
k_5:	k_5:	k_5:	k_5:o_9、o_{10}
k_6:o_1、o_5	k_6:o_4	k_6:o_8	k_6:
k_7:	k_7:	k_7:	k_7:o_7、o_9
k_8:	k_8:o_4、o_6	k_8:o_3	k_8:o_9

图 1 中，sf 表示该结点对应的签名文件，结点的签名文件是由该结点中所有对象的签名文件进行 or 操作产生。在 STTR-Tree 中，假设签名文件为一串

8 位的二进制码,通过设定的 hash 函数将关键词映射到每一位二进制码中。如果二进制码中的位为 1,则表示该位包含对应的关键词,若二进制码中的位为 0,则表示该位不包含对应的关键词。例如,在 STTR-Tree 中,假设 o_1、o_2、o_5 的签名文件分别为 10010000、01000100、00010100,将 o_1、o_2、o_5 的签名文件进行 or 操作,生成结点 N_1 的签名文件 11010100。同理,其它结点以同样的方式生成相应的签名文件。

算法在执行查询过程时,首先查询关键词与结点包含的关键词进行匹配,将查询关键词的签名文件与结点包含的签名文件执行 and 操作,若两个二进制签名文件执行 and 操作的结果与查询关键词生成的二进制签名文件相同,则表示该结点包含查询关键词,反之则不包含。例如,查询关键词生成的签名文件为 00000101,对于 STTR-Tree 根节点,将查询签名文件与根结点的签名文件进行 and 操作,00000101 and 11011111 = 00000101,此结果表示根结点包含查询关键词。

NumTextP 表示指向该结点的数值型信息的指针,结点的数值型信息同时包含了该结点的所有对象的数值型信息。叶子结点的数值型信息见表 4。

表 4 数值型信息

Tab. 4 Numerical information

NumTextP-N_1			NumTextP-N_2			NumTextP-N_3			NumTextP-N_4		
对象名称	人均价格	用户评分	对象名称	人均价格	用户评分	对象名称	人均价格	用户评分	对象名称	人均价格	用户评分
O_1	77	8.5	O_4	55	8.5	O_3	95	9.2	O_7	66	7.5
O_2	65	8.8	O_6	71	7.6	O_8	110	9.5	O_9	52	7.1
O_5	81	9							O_{10}	73	8.2

非叶子结点主要包含以下信息:该结点所有子结点的最小边界矩形(mbr)、指向该结点的子结点指针(cnp)、该结点包含的所有子结点时间段的并集(Time)、该结点对应的签名文件(SF),结点的签名文件是由所有子结点的签名文件进行 or 操作产生的。结点 N_1、N_2、N_5包含的时间段信息见表 5。

表 5 结点的时间段信息

Tab. 5 Time period information of nodes

结点	时间段
N_1	6:00-20:00
N_2	10:00-17:00、18:00-24:00
N_5	6:00-24:00

4 算法描述

本节根据 STTR-Tree 索引提出了 TSTKSQ 的裁剪策略和算法。TSTKSQ 算法在遍历 STTR-Tree 索引时,先判断结点是否在查询范围之内,然后将结点包含的关键词和时间段信息与查询关键词和时间段信息进行相交判定;算法从空间、关键词和时间 3 个属性对空间数据集上的对象进行过滤;当算法遍历至叶子结点时,将筛选出关键词相关和时间相关的对象进行数值型信息支配和基于时间的空间文本关键词支配关系判断,最终获取查询结果集。

4.1 裁剪策略

TSTKSQ 算法在遍历 STTR-Tree 索引时,对结点采用如下裁剪策略:

(1)若查询关键词与结点包含的关键词不相交,则不必进行时间段相交的判断,直接将结点进行剪枝。

(2)若查询时间段与结点包含的时间段不相交,则不必进行关键词相交的判断,直接将结点进行剪枝。

(3)若同时满足查询关键词与结点包含的关键词相交,以及查询时间段与结点包含的时间段相交,则对其子结点进行重复判断,直到筛选出满足条件的候选对象,否则将结点进行剪枝。

在基于 STTR-Tree 的查询算法和裁剪算法中,本文使用优先队列对候选集 C 和结果集 R 进行维护,优先队列中的对象按照 TSKR 的非递减顺序出队列。

定理 1[3] 在按照 TSKR 的非递减顺序出队列的优先队列中,首个出队列的对象 o 必为 skyline 对象。

定理 2 给定数据集中的任意两个对象 o_i、o_j,如果 o_i 与 o_j 之间不存在数值型信息支配,则 o_i 与 o_j 之间也不存在基于时间的空间文本关键词支配。

证明 由于 o_i 与 o_j 之间不存在数值型信息支配关系,根据定义 2 可知,o_i 与 o_j 之间不能同时满足 $o_i \prec_{NI} o_j$ 且 $\mathrm{TSKR}(q,o_i) \leqslant \mathrm{TSKR}(q,o_j)$,所以 o_i 与 o_j 之间也不存在基于时间的空间文本关键词支配。

例如，表 1 中的对象 a 和 b，由于 a 在用户评分这一属性上支配 b，而 b 在人均价格这一属性上支配 a，所以二者不存在数值型信息支配关系，因此二者也不存在基于时间的空间文本关键词支配关系。

定理 3[3]　在按照 TSKR 的非递减顺序出队列的优先队列中，若已出队列的对象为 o，在 o 之后出队列的任意对象为 o'，必有 $o' \not\prec_{TSTK} o$。

定理 4　在按照 TSKR 的非递减顺序出队列的优先队列中，设先出队列的对象为 o，后出队列的对象为 o'，若 $o_i \prec_{NI} o'$，则 $o \prec_{TSTK} o'$。

证明　根据优先队列的性质可知，TSKR(q,o) ≤ TSKR(q,o')，根据定义 2 可知，$o \prec_{TSTK} o'$。

例如定义 2 中的例子，对象 b 和 d 的 TSKR 分别为 4、6，因此先出队列的对象为 b，后出队列的对象为 d，又因为 $b \prec_{NI} d$，所以 $b \prec_{TSTK} d$。

在基于 STTR-Tree 的 TSTKSQ 算法中，本文对候选对象采用如下裁剪策略：

按照 TSKR 的非递减顺序出队列的优先队列中，设当前出候选集队列的对象为 o，当前结果集中的任一对象为 sp，若 $sp \prec_{NI} o$，则裁剪 o，否则将对象 o 放入结果集中。

证明　根据优先队列的性质可知，TSKR(q, sp) ≤TSKR(q,o)，若 $sp \prec_{NI} o$，根据定义2可知 sp 基于时间的空间文本关键词支配 o，此时对象 o 可以被裁剪，反之，若 sp 与 o 之间不存在数值型信息支配关系，根据定理 2，sp 与 o 之间也不存在基于时间的空间文本关键词支配，所以 o 为 *skyline* 对象，放入结果集中。

4.2　算法

算法 1　TSTKSQ 算法

输入　查询点 q、查询关键词 qk、查询时间段 qtq、查询范围 r、STTR - Tree 索引、空间对象点集 O

输出　查询结果集 R

```
1   R = ∅; C = ∅;   //R 存放查询结果集,C 存放候选集
2   While not Stack.is Empty( ) do   //以深度优先遍历索引
3       N = Stack.pop( );
4       If d(q,N) < r   // 若结点在查询范围之内
5       If qk∩Nk ≠ ∅//若查询关键词与结点包含的关键词相交
6       If qtq ∩ Ntq ≠ ∅// 若查询时间段与结点包含的时间段相交
7       If N.is Leaf( ) then
8       For each o in N do
9       If qk ∩ ok ≠ ∅
10      If qtq ∩ otq ≠ ∅
11      C ← NewPriorityQueue; //按照 TSKR 的非递减顺序初始化优先队列
12      C.Enqueue(o);   // 将对象 o 放入候选集优先队列中
13      Else
14      Stack.push (N.ChildNode);   // 将孩子结点进栈
15  end While
16  R ← NewPriorityQueue;   //按照 TSKR 的非递减顺序初始化优先队列
17  R = dominateCompting(q,C);   //对候选集中的对象进行支配计算
18  return R;
```

算法 1 是基于时间的空间文本关键词 skyline 查询的具体过程。第 2-3 行以栈的方式维护索引，第 4-7 行对查询范围内的结点进行判断，筛选出查询关键词与结点包含关键词相交以及查询时间段与结点包含时间段相交的结点，直到遍历至叶子结点。第 8-12 行遍历叶子结点，筛选出查询关键词与对象包含关键词相交以及查询时间段与对象包含时间段相交的对象，将对象放入 TSKR 的非递减候选集优先队列中。第 16-18 行将候选集中的对象进行支配计算，把不被支配的对象放入结果集队列中。由于第 17 行中 dominateCompting( ) 算法需要判断所有候选对象间的支配关系，导致算法整体查询效率下降，因此需要加入高效的裁剪策略来提升查询效率。

算法 2　dominateComputing( ) 算法

输入　候选集 C、查询点 q、查询关键词 qk、查询时间段 qtq

输出　查询结果集 R

```
1   R← getCFirst( );   //将 C 中首个出队列对象放入结果集中
2   While not C. isEmpty( ) do
3       o = C. Dequeue( );
4       If sp NumericTypeDominate o   //判断结果集中的对象 sp 是否数值型信息支配对象 o
5       contine;
6       Else
7       insert o into R
8   end While
```

9 return R

算法 2 是判断候选集对象间支配关系的裁剪算法。第 1 行是将候选集中首个出队列对象放入结果集中。第 2-3 行若候选集队列非空时,依次取出候选集中的对象进行判断,第 4-7 行若候选对象被结果集中的对象基于时间的空间文本关键词支配则删除候选对象,否则将对象放入结果集中。

以图 1、表 2~表 5 中包含的数据为例,假设查询关键词为 k_1、k_2、k_4,生成对应二进制签名文件为 110 100 00,查询时间段为 10:00-12:00,默认各结点在查询范围之内,并且数值型属性的要求为人均价格低、用户评分高。具体查询过程如下:

首先,筛选出查询关键词和查询时间与对象的关键词和时间相交的候选对象。从根节点开始,将查询二进制签名文件与结点包含的签名文件进行 and 操作,110 100 00 and 110 111 11 = 110 100 00 表示根节点包含查询关键词,并且根节点的时间段包含查询时间段,然后对其孩子结点进行重复判定,110 100 00 and 110 101 01 = 110 100 00 表示结点 N_5 包含查询关键词,并且结点 N_5 的时间段包含查询时间段,110 100 00 *and* 100 111 11 = 100 100 00 表示结点 N_6 不包含查询关键词,则对 N_6 及其孩子结点进行裁剪,不必进行后续判定提高了查询效率,继续对结点 N_5 的孩子结点 N_1、N_2 进行判断,110 100 00 and 110 101 00 = 110 100 00 表示结点 N_1 包含查询关键词,并且结点 N_1 的时间段包含查询时间段,110 100 00 and 010 101 01 = 010 100 00 表示结点 N_2 不包含查询关键词,直接进行剪枝,此时得到叶子结点 N_1 中的对象 o_1、o_2、o_5,由于 o_1 的时间段与查询时间段不相交,删除 o_1,而 o_2、o_5 满足查询关键词与查询时间都相交,此时得到候选集对象 o_2、o_5。

之后,判断候选对象间的支配关系。假设查询点与对象 o_2、o_5 的空间距离分别为 1、2,根据计算 o_2、o_5 的 TSKR 分别为 1.2、2.4,将 o_2、o_5 按照 TSKR 的非递减顺序放入候选集队列中,将第一个出队列的对象 o_2 直接加入结果集中,然后 o_5 出队列,由于 o_2 与 o_5 间不能构成数值型信息支配,因此将 o_5 加入结果集中,此时遍历完所有候选对象,得到最终结果集 $\{o_2、o_5\}$。

5 实验结果与分析

实验采用的硬件设备为 64 位 Windows 10 操作系统,Intel(R) Core(TM) i5-7200U CPU @ 2.50 GHz处理器,8 G 内存;采用 Java 语言实现算法,集成开发环境为 IntelliJ IDEA Community Edition 2021.1.3,JDK 版本为 11.0.11。

实验数据来源于 yelp 网站的开源数据集,该数据集中包括克利夫兰、多伦多等 11 个城市 150 346 个商户的信息。实验将数据集中的经纬度作为对象的空间位置信息,价格、星级等作为对象的数值型信息,商户的分类作为对象的关键词信息,营业时间作为对象的时间信息。通过是否使用裁剪策略 TSTKSQ 与 NTSTKSQ 测试算法的有效性,每次测试均取相同环境下 10 次测试的平均值为最终结果。

5.1 查询关键词数量的影响

为了测试查询关键词的数量对算法的影响,设置数值型属性为 2 维,查询点的空间位置和查询时间段固定不变,查询关键词 1~5 个。查询关键词数量变化对算法的影响如图 2 所示。

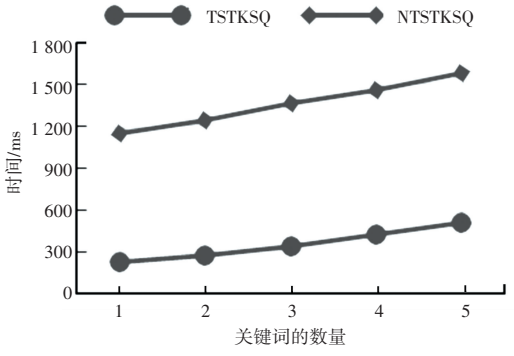

图 2 查询关键词数量的影响

Fig. 2 Impact of the number of query keywords

从图 2 中可知,随着查询关键词数量的增加,算法整体的运行时间也不断增加。对于不使用裁剪策略 NTSTKSQ 进行查询时,算法需要遍历所有数据集中的对象,将查询关键词与每个对象包含的关键词一一比较,直到筛选出所有包含查询关键词的对象,而随着查询关键词数量的增加,包含查询关键词的对象也越来越多,因此整体查询时间呈上升趋势,而使用裁剪策略 TSTKSQ 进行查询时,算法的查询时间明显少于使用裁剪策略 NTSTKSQ 的查询时间。由于使用裁剪策略 TSTKSQ 时,算法根据 STTR-Tree 结点的签名文件进行操作时,提前裁剪了不包含查询关键词的结点,不必进行后续的判断,因而极大地提升了算法的执行效率。

5.2 数值型属性维度的影响

为了测试数值型属性维度对算法的影响,设置了 2 个查询关键词,查询点的空间位置和查询时间段固定不变,数值型属性维度从 1 维变化到 5 维。数值型属性维度的变化对算法的影响如图 3 所示。

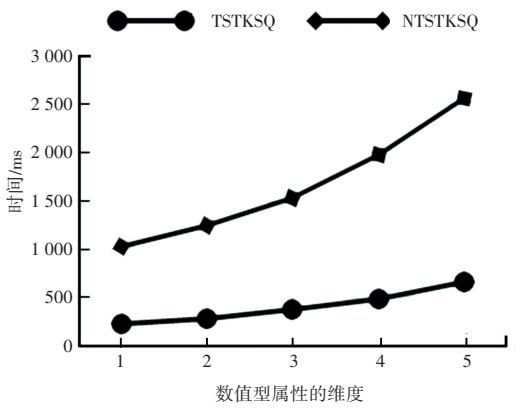

图 3　数值型属性维度的影响

Fig. 3　Impact of numerical attribute dimensions

从图中可知，随着数值型属性维度的增加，算法整体的运行时间呈上升趋势。对于不使用裁剪策略 NTSTKSQ 进行查询时，算法需要遍历所有数据集中的对象，判断对象间的数值型信息支配关系，而随着数值型属性维度的增加，对象间的数值型信息支配判断次数也不断增加，因此整体查询时间也不断增加；而对于使用裁剪策略 TSTKSQ 进行查询时，算法的查询时间明显少于不使用裁剪策略 NTSTKSQ 的查询时间。因为使用裁剪策略 TSTKSQ 时，算法过滤了大部分查询关键词和查询时间不匹配的对象，大大减少了候选集对象间数值型信息支配次数的判断，缩短了整体支配判定的时间。

5.3　查询时间段大小的影响

为了测试查询时间段大小对算法的影响，设置查询关键词为 2 个，数值型属性为 2 维，查询点的空间位置固定不变，查询时间段不断增加。查询时间段的变化对算法的影响如图 4 所示。

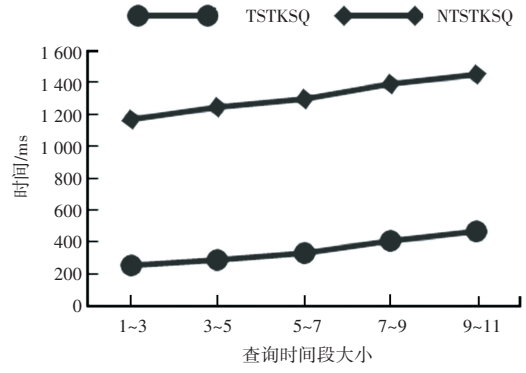

图 4　查询时间段大小的影响

Fig. 4　Impact of query time period size

从图中可知，随着查询时间段不断增加，算法整体的运行时间呈上升趋势。对于不使用裁剪策略 NTSTKSQ 进行查询时，算法需要遍历所有数据集中的对象，将查询时间段与每个对象包含的时间段一一比较，直到筛选出所有包含查询时间段的对象，而随着查询时间段大小的增加，包含查询时间段的对象也越来越多，需要比较的次数也不断增加，因此整体查询时间呈上升趋势，而对于使用裁剪策略 TSTKSQ 进行查询时，算法的查询时间明显少于不使用裁剪策略 NTSTKSQ 的查询时间。因为使用裁剪策略 TSTKSQ 时，算法根据 STTR-Tree 结点包含的时间信息进行匹配时裁剪了不包含查询时间段的结点，算法因此过滤了大部分不包含查询时间段的对象，极大地减少了查询时间。

6　结束语

为了解决用户在实践中更多偏好查询的需求，本文提出了基于时间的空间文本关键词 skyline 查询 TSTKSQ，同时考虑了空间距离、数值型信息、关键词和时间等 4 个属性，并构建了相应的空间索引 STTR-Tree，同时设计了时空关键词相关性评价函数，以此来衡量一个对象的优劣程度，然后，提出了结点和对象的裁剪策略，并在此基础上提出了高效的 skyline 查询算法。最后，在真实数据集上进行测试，实验结果表明所提算法能够有效地解决基于时间的空间文本关键词 skyline 查询问题。之后的工作考虑将 TSTKSQ 应用于道路网中，进一步研究道路网中 TSTKSQ 问题的解决方法。

参考文献

[1] BÖRZSÖNYI S, KOSSMANN D, STOCKER K. The skyline operator[C]//Proceedings 17th International Conference on Data Engineering. Heidelberg, Germany: IEEE, 2001: 421-430.

[2] 李星罗，秦小麟，王宁，等. 空间多关键词 Skyline 查询算法[J]. 小型微型计算机系统，2019，40(10)：2175-2181.

[3] 郭莎莎，李爽，阎红灿. 已知时间的空间文本 skyline 查询[J]. 计算机工程与应用，2020，56(24)：59-65.

[4] 余未，郑吉平，王海翔，等. 空间 Skyline 查询处理：应用、研究与挑战[J]. 计算机科学，2017，44(2)：1-16.

[5] MAO R. Spatial skyline query problem in Euclidean and road-network spaces[D]. Canada: Simon Fraser University, 2020.

[6] 李松，窦雅男，郝晓红，等. 道路网环境下 K-支配空间 Skyline 查询方法[J]. 计算机研究与发展，2020，57(1)：227-239.

[7] CAI Zhi, CUI Xuerui, SU Xing, et al. Continuous road network-based skyline query for moving objects[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(12): 7383-7394.

[8] SON W, STEHN F, KNAUER C. Top-k manhattan spatial skyline queries[J]. Information Processing Letters, 2017, 123: 27-35.

[9] BAVIRTHI S S, SUPREETHI K P. An approach for combining spatial and textual skyline querying using indexing mechanism[J]. Turkish Journal of Computer and Mathematics Education, 2021, 12(11): 672-680.

（下转第 49 页）