Diversity in spatio-textual data objects

Andreas Alamanos[1], Akrivi Vlachou[1]

[1]Department of Information and Communication Systems Engineering (ICSD), University of the Aegean, Karlovasi, Greece

Abstract

In recent years, more and more services rely on the user's location to provide relevant information to the user. Many of the location-based services (LBS) allow users to search for points of interest (POIs), such as restaurants, hotels, etc., based on their preferences and the distance from them. In such applications, the queries posed by the users actually include spatial and textual information. In this paper, we address the problem finding the most popular points of interest based on a set of users queries but at the same time our result set should be of high diversity in order to represent the preferences of all users. We first provide an appropriate problem definition, so that the selected points of interest are dissimilar to each other but also popular for the users. We evaluate experimentally our approach and our experimental evaluation shows that in all cases our approach succeeds to retrieve objects of high diversity.

Keywords

Spatio-textual Queries, Diversity, Top-k Queries

1. Introduction

Nowadays, many applications such as location-based services (LBS), allow the users to pose queries based on their location. Usually, in order to avoid overwhelming the users with many object suggestions, a restricted set of k objects is presented to the user. In many applications, users express their preferences by providing textual description (keywords) and k objects are retrieved and sorted based on the distance to the user and the textual similarity. Such queries are know as spatial-keyword search queries and location-based services (LBS) allow users to search for points of interest (POIs), such as restaurants, hotels, etc., by processing spatial-keyword search queries.

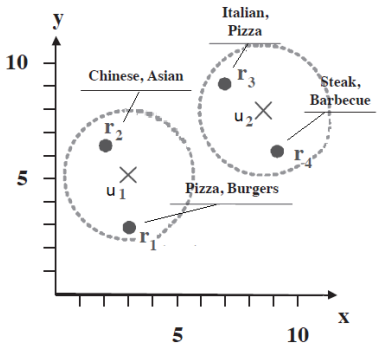


Figure 1: Example.

Example. Consider for example, a tourist that looks for a "nearby Italian restaurant that serves pizza". Figure 1 depicts a spatial area containing user locations (query points) and restaurants (points of interest). Each restaurant has textual information in the form of keywords extracted from its menu, such as pizza or steak, which describes additional characteristics of the restaurant. The tourist also specifies a spatial constraint (in the figure depicted as a range around his location) to restrict the distance of restaurants to his position. Obviously, the best option for a tourist u_2 that poses the aforementioned query is the restaurant r_3, because its within a given range and contains the given keywords. On the other hand, for the tourist u_1, the best option is r_1. In the general case, many different users with different locations and different preferences expressed as keywords are using location based services.

Even though spatial-keyword queries have been studied before [1, 2, 3], in this paper, we address a different problem. The focus of this paper is to provide an approach to analyze the preferences of a set of users that are indirectly expressed by the queries they have been posed. The points of interests that have been retrieved and presented to the users are popular for these users. Since many different queries may have posed by the same or different users, we focus on selecting a restricted set of m points of interests that are popular and at the same time are diverse, in order to cover the preferences of as many users as possible.

In this paper, we assume that there exists a query log file and we formulate a novel approach (Diversified m Selection Problem $DmSQ$) that retrieves a set of m points of interest of high diversity. The concept of diversification has been introduced in several systems, to avoid presenting to the user similar objected that may fail to trigger his interest. Our primary objective is to select a set of spatio-textual objects that are popular based on the preferences of a set of users, but at the same time cover their interests. Thus, we consider as candidate objects, all objects that have been retrieved and presented to the users through the queries they have posed. This objects are considered popular, since that objects match the users preferences. Then, based on the similarity of the candidate objects, our goal is to select those objects that are dissimilar to each other, i.e, maximize diversity of the retrieved set.

To this end, we first define the notion, of similarity and diversity for spatio-textual objects (Section 3) and then, formulate our problem statement (Section 3.3) and provide an appropriate algorithm (Section 4). In our experimental evaluation (Section 5), we study the performance of our approach using varying number of queries, number of retrieved objects of interest (k) and number of querying keywords. We compare our approach against a naive approach which takes into account only the popularity of the objects. Our experimental evaluation shows that in all cases the Diversified m Selection Problem $DmSQ$ succeed to retrieve objects of high diversity.

Published in the Proceedings of the Workshops of the EDBT/ICDT 2024 Joint Conference (March 25-28, 2024), Paestum, Italy

icsdd22009@icsd.aegean.gr (A. Alamanos); avlachou@aegean.gr (A. Vlachou)

2. Related Work

The spatial keyword query is extensively studied. Surveys of the different proposed approaches are provided in [3, 4].

In [2], a categorisation of the indexing methods is provided, based on the spatial or the text prioritisation of the underlying structures, as text-first or spatial-first combination of spatial and text indices. The authors of [5] introduce the spatial-first IR^2-Tree indexing method which embodies superimposed text signatures in a plain IR-Tree [6] for solving the top-k spatial keyword searching. The authors of [7] attempted to resolve efficiency issues of the IR^2-Tree, by providing a spatial inverted index. A text-first index for top-k spatial keyword queries, named S2I, was introduced in [8]. The algorithm maps each keyword to a different aggregate R-tree which stores the objects with the given term. An experimental study for 12 geo-textual indexing structures was carried out in [3]. Their results offer a usage manual for the examined algorithms.

A variant of spatial-keyword search is proposed in [9]. The process, called top-k MULTI query, involves expanding the top-k query to encompass additional data types beyond text and spatial categories. This is done in a manner that ensures the original top-k spatial-keyword search is a specific case within this broader process.

The outcomes of diversification in spatial queries for identifying top-k results have drawn notable attention in the literature, with several solutions focusing on an incremental way of retrieving the results. Diversification is mostly bound on the dissimilarity in the context of content, novelty and coverage [10]. Two greedy algorithms for query result diversification are introduced in [11], Greedy Marginal Contribution (GMC) and Greedy Randomized with Neighborhood Expansion (GNE). The former incrementally builds the results by selecting the element with the highest maximum marginal contribution, whilst the latter diversifies the results by choosing a random element, among the top ranked ones, is chosen.

The methodology employed in [12] involves utilizing a graph representation and diversification approach on top-k results. The authors propose a set of new functions aiming to more efficient search on big graphs. The objectives outlined in [13] align with the aforementioned goals; nevertheless, the authors place particular emphasis on querying data initially presented as a knowledge graph and subsequently diversifying the sub-graphs.

The authors of [14] aim to accomplish the diversification of the results based on the users' known preferences. In pursuit of this objective, a diversity is defined, where each object is represented by its reverse top-k result set, and retrieves the m objects that maximize their diversity value. In [15], the authors diversify their results using normalized relevance, coverage, and execution time evaluation metrics. They introduce the PrefDIv algorithm, an incremental method that eliminates similar items originally retrieved until it reaches the threshold k.

In [16] a diversification framework is presented that considers spatial and contextual similarity together with contextual and spatial proportionality. Proportionality is obtained by evaluating the characteristics of the retrieved items within designated categories and ensuring a proportional representation of these objects. In [17] an alternative approach is followed combining spatial proximity with social diversity.

3. Problem Statement

In this section, we provide the necessary definitions and our novel problem statement.

3.1. Preliminaries

Let $O = \{o_1, o_2, .., o_j\}$ a set of spatio-textual objects, defined by a spatial and a textual part. The primary objective of the spatial-keyword search is to retrieve the k objects that match the given preferences and are ranked by the minimal distance from a given query location. Different query types have been proposed in the related literature [3]. Our diversity approach can be applied on any query type, but for sake of simplicity we assume that the retrieved objects are described by at least one query term, while the ranking is based on spatial distance.

Definition 1 (Keyword kNN Query (KNQ))**.** *A KNQ $q = <x, y, t, k>$ takes four parameters, where $q.x$, $q.y$ define the coordinates of a spatial point, $q.t$ is a set of keywords, and $q.k$ is the number of objects to retrieve. The result Q of a KNQ query is a set of k objects such that $\forall o \in Q$ ($\nexists o' \in (O - Q : dist(o', q) \leq dist(o, q) \wedge \{q.t \bigcap o'.t\} \neq \emptyset)$, where $dist$ a spatial distance function.*

Thus, the result set of a KNQ query is a set of k objects o_i, such that at least one query term $q.t$ is contained in the textual part of each object and the objects are ranked according to their distance to the query location.

Given a set of l query result sets $Q = \{Q_1, Q_2, .., Q_l\}$ that correspond to the preferences of the users that pose queries, the goal is to retrieve m objects that are interesting for all the l users. Thus, the candidate objects are the objects retrieved by their queries and a subset of m are selected based on their dissimilarity in order to provide a set of high diversity.

Even though in the following we assume that KNQ queries are posed, our approach can support any spatio-textual query[3], since only the query result sets are needed to be stored and not the queries themselves. This is also a benefit as far as privacy issues are concerned since the actual query q that may contain sensitive information, such as the user location, does not need to be stored.

3.2. Motivating Example

In Table 1 we depict a small dataset[1] containing 12 restaurants in Athens, GA, USA. Each tuple corresponds to a restaurant and has an artificial ID (i), a unique restaurant identifier (RID), the restaurant's name, its longitude and latitude and a set of keywords that describe the cuisine of the restaurant. During the example we will refer to each object i as o_i. Given a query q_j and its query result Q_j, we define as ${rank_i}^j$ the ranking position of an object o_i.

Let us assume that a user is at the location (-83.35, 33.95) looks for the 3 closest restaurants that serve American cuisine. Thus, the user poses a query q_1 = (-83.35, 33.95, $\{American\}$, 3) and the result set is Q_1 ={DePalma's Italian Cafe - East Side (o_9), DePalma's Italian Cafe - Downtown (o_8), Last Resort Grill (o_1)}. Thus, it holds that ${rank_9}^1 = 1$, ${rank_7}^1 = 2$ and ${rank_1}^1 = 3$. Table 2 shows the result sets of three different queries.

[1] https://www.kaggle.com/datasets/shrutimehta/zomato-restaurants-data/

i	RID	Name	Longitude	Latitude	Cuisines
1	17293281	Last Resort Grill	-83.378273	33.957999	American. Southern. Southwestern
2	17293301	Mama's Boy Restaurant	-83.3654	33.9535	Southern
3	17293409	Sr. Sol 1	-83.4293	33.9652	Mexican
4	17293163	Choo Choo Eastside	-83.3389	33.9259	Japanese. Korean
5	17293228	The Grill	-83.375523	33.958198	Breakfast. Burger. Sandwich
6	17293880	Big City Bread Cafe	-83.384004	33.959392	Breakfast. Sandwich
7	17293169	Clocked	-83.3797	33.9584	American. Burger. Sandwich
8	17293186	DePalma's Italian Cafe - Downtown	-83.373596	33.958112	American. Italian. Pizza
9	17293180	DePalma's Italian Cafe - East Side	-83.33995	33.924275	American. Italian. Pizza
10	17293205	Five Ten	-83.3872482	33.9415545	American
11	17293229	Grit	-83.381625	33.960112	International. Southern. Vegetarian
12	17293422	Transmetropolitan	-83.3764	33.9584	Italian. Pizza. Sandwich

Table 1
Dataset Example.

Query	Top-1	Top-2	Top-3
$q_1 = (-83.35, 33.95, \{American\}, 3)$	DePalma's Italian Cafe East Side (o_9)	DePalma's Italian Cafe Downtown (o_8)	Last Resort Grill (o_1)
$q_2 = (-83.25, 33.96, \{Italian, Pizza\}, 3)$	DePalma's Italian Cafe Downtown (o_8)	DePalma's Italian Cafe East Side (o_9)	Transmetropolitan (o_{12})
$q_3 = (-83.38, 33.93, \{Breakfast, Vegetarian\}, 3)$	The Grill (o_5)	Big City Bread Cafe (o_6)	Grit (o_{11})

Table 2
Example of Query Result Sets.

In our scenario, the aim is to select m restaurants that are popular but also diverse in the sense that they cover the interest for all users, whose preferences are expressed by the queries they have posed. Assuming $m = 2$, based on the result sets in Table 2 we could conclude that the most popular objects are {DePalma's Italian Cafe - East Side (o_9), DePalma's Italian Cafe - Downtown (o_7)}, but these objects fail to take into account diversity.

Thus, a naive way is to find the m most popular objects by selecting the m objects that have the highest $\sum_{\forall O_j}(|O_j| - rank_i{}^j)$. This means to select the objects that are ranked high in as many as possible query result sets. Even though this approach selects popular objects, it fails to handle diversity. Thus, objects that are highly ranked in similar queries may be selected, while other users may not represented by the select objects. In our example, the third user is not interested in any restaurant of the selected set {DePalma's Italian Cafe - East Side (o_9), DePalma's Italian Cafe - Downtown (o_8)}.

Definition 2 (Naive m Selection Query ($NmSQ$))**.** *Given a set of queries $\{q_i\}$ and an integer m, the $NmSQ$ set is a set that satisfy the following two conditions:*

1. $NmSQ \subseteq \bigcup Q_i$ *and* $|NmSQ| = m$.
2. *For any* $o_j \in NmSQ$ *and* $o_z \in \bigcup Q_i - NmSQ$, $rank_j \geq rank_z$.

The naive selection query does not take into account diversity. In following we address this problem.

3.3. Problem Statement

Diversifying query result sets is an important problem [12] since in many real-life applications the users only inspect a small set of m objects. In the following, we first define the notion of similarity of two spatio-textual objects o_i, o_j.

Definition 3 (Similarity of spatio-textual objects o_i, o_j:)**.** *The similarity $sim(o_i, o_j)$ of two spatio-textual objects o_i, o_j is defined as:*

$$sim(o_i, o_j) = \alpha * SDist(o_i, o_j) + (1 - \alpha)TRel(o_i, o_j),$$
$$SDist(o_i, o_j), TRel(o_i, o_j) \in [0, 1]$$

where $SDist(o_i, o_j)$ measures the spatial similarity and is defined as $SDist(o_i, o_j) = \frac{(max\_dist - dist(o_i, o_j))}{max\_dist}$, $dist(o_i, o_j)$ is a spatial distance function and max_dist the maximum distance between any two objects in O. $TRel(o_i, o_j)$ measures the textual similarity, such as Jaccard index. Parameter α defines the relative importance of the spatial and textual similarity, otherwise it is set to 0.5.

Example. Given the objects o_1 and o_2 of Table 1, spatial similarity is computed based on the Harversine distance. The Harversine distance between o_1 and o_2 is $dist(o_1, o_2) = 1.4$ km, while the maximum distance among all objects in the example set is $max\_dist = 10.06$ km. Consequently, the spatial similarity is $SDist(o_1, o_2) = \frac{(10.06 - 1.4)}{10.06}$, resulting in a value of 0.86. Similarly, the textual similarity is calculated by assessing the intersection and union of two sets of keywords ($o_1.t$,$o_2.t$) where the term "Southern" is the common element. The textual similarity component is then determined as: $TRel(o_1, o_2) = \frac{|o_i \cap o_j|}{|o_i \cup o_j|} = \frac{1}{3} \approx 0.333$. Subsequently,the overall similarity is computed as the weighted sum of the spatial and textual similarity components: $sim(o_1, o_2) = \alpha * 0.86 + (1 - \alpha) * 0.333 = 0.59$ ($\alpha = 0.5$).

We extend the notion of spatial-textual objects similarity for a set of objects S, $\mathcal{S} = \{o_1, o_2, ..o_j\}$.

Definition 4 (Similarity of points $o_1, o_2, ..o_j$)**.** *We define the similarity $\mathcal{S}$ for a set of j objects S, $\mathcal{S} = \{o_1, o_2, ..o_j\}$ as*

$$sim(\mathcal{S}) = \alpha * SDist(\mathcal{S}) + (1-\alpha)TRel(\mathcal{S}),$$
$$SDist(\mathcal{S}), TRel(\mathcal{S}) \in [0,1]$$

where $SDist(\mathcal{S}) = \frac{(max\_dist - Avg(dist(o_i,..o_j)))}{max\_dist}$, where $Avg(dist(o_i,..o_j))$ is used to denote the average pairwise distance between the j objects, and $TRel(\mathcal{S})$ a textual similarity function such as the extended version of the Jaccard index [18].

In order to avoid overwhelming the users with many object suggestions, in the majority of applications a restricted set of m objects is presented to the user. The concept of diversification has been introduced in several systems, to avoid presenting to the user similar objected that may fail to trigger his interest. In the current paper, our primary objective is to select a set of spatio-textual objects that are popular based on the preferences of a set of users, but at the same time cover all their interests. Thus, we consider as candidate objects a collection of spatio-textual query result sets that express the user preferences and the selected objects maximize their dissimilarity, i.e, diversity.

Definition 5 (Diversity of a set $O = \{o_1, o_2, ..o_m\}$). *We define as diversity of a set* $O = \{o_1, o_2, ..o_m\}$

$$div(O) = 1 - sim(\{o_1, o_2, ..o_j\})$$

Example. The assessment of object similarity of objects o_1, o_2 and o_3 (Table 1) involves calculating the average Haversine spatial distance, yielding 4.7 km. The spatial similarity is then computed as $SDist(\{o_1, o_2, o_3\}) = \frac{(10.06-4.7)}{10.06}$, resulting in 0.52. Similarly, the textual similarity is calculated by assessing the intersection and union of the three keywords sets. The intersection, yields zero, thus $TRel(\{o_1, o_2, o_3\}) = 0$. This indicates a complete dissimilarity, as no shared elements exist among the sets. Subsequently,the overall similarity is $sim(\{o_1, o_2, o_3\}) = 0.26$. Thus, the is $div(\{o_1, o_2, o_3\}) = 1 - 0.26 = 0.74$

Definition 6 (Diversified m Selection Problem $DmSQ$). *Given a set of spatio-textual points* $O = \{o_1, o_2, ..\}$, *and an integer* m *where* $1 \leq m \leq |O|$, *the diversified* m *selection problem result set, denoted as* $DmSQ$, *is a set of objects that satisfy the following two conditions:*

1. $DmSQ \subseteq O$ *and* $|DmSQ| = m$
2. $\nexists DmSQ'$ *such that* $DmSQ' \neq DmSQ$, $|DmSQ'| = m$ *and* $div(DmSQ') > div(DmSQ)$.

The above definition ensures that the diversity $div(DmSQ)$ is maximised compared to all other subset of size m.

4. Algorithm

In order to solve the Diversified m Selection Problem $DmSQ$ we propose a greedy algorithm (Algorithm 1). Given a set of candidate objects $\mathcal{C} = \bigcup Q_j$ (i.e., the objects that belong to the query result sets), our algorithm first inspects all pairs of candidate objects and computes their dissimilarity $div()$. From those pairs it selects the pair with the higher dissimilarity. Thereafter, it inspects the triples that contain the two selected objects and one of the remaining candidate objects, and selects the objects that result in the higher $div()$. This processes is repeated until m objects are selected.

Algorithm 1 *Greedy algorithm for the Diversified* m *Selection Problem* $DmSQ$

input: Set of candidate objects $\mathcal{C} = \bigcup Q_j$
m number of returned objects
output: The set $DmSQ$ of m diverse objects

```
1:  M ← ∅, d = 0
    {Initialization step}
2:  for i = 1...|C| do
3:    for j = i + 1...|C| do
4:      if div(o_i, o_j) ≥ d then
5:        M = {o_i, o_j}
6:        d = div(o_i, o_j)
7:      end if
8:    end for
9:  end for
    {Selection of m − 2 objects}
10: while |M| < m do
11:   T ← ∅, d = 0
12:   for all o ∈ {C − M} do
13:     if div(T) ≥ d then
14:       T = M ⋃ {o}
15:       d = div(T)
16:     end if
17:   end for
18:   M ← T
19: end while
20: return M
```

Algorithm 1 takes as input the result sets of the user's queries and the parameter m that defines the number of returned objects. In the initiation step (lines 2–9), all objects undergo pairwise cross-comparison with each other with respect to their dissimilarity and the most promising pair of objects is selected. Thereafter (10–19), one point is selected in each repetition in such a way that the dissimilarity is maximized. This process is repeated until m points are selected and finally, the selected set $\mathcal{M}$ is returned (line 20).

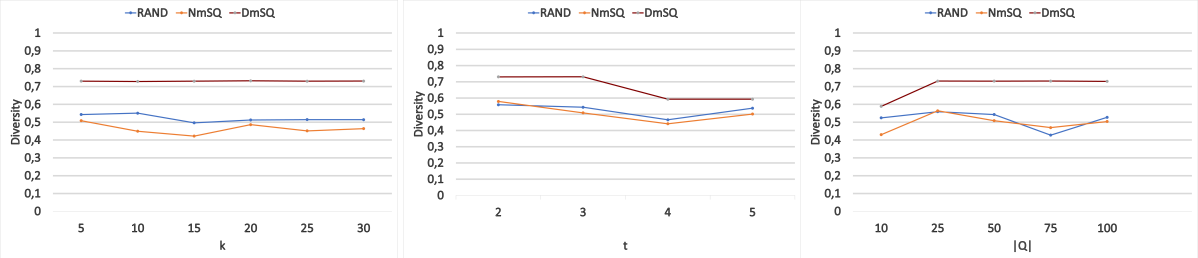
For the initiation step, our algorithm performs $|\mathcal{C}|^2$ comparisons, while for the remaining steps the comparisons are $m * |\mathcal{C}|$, which results in a complexity of $O(|\mathcal{C}|^2)$. In order to reduce the algorithmic complexity our algorithm could avoid the first step and select a random point as the first selected candidate. Obvious, the diversity of the selected objects is smaller in this case. Alternative, a spatio-textual index structure [8, 3, 19] could be used to speed up the comparison and to prune pairs of candidates that cannot lead to high diversity.

5. Experimental Evaluation

In the following, we first present our experimental setup and then we describe our experimental results

5.1. Experimental Setup

Given a dataset, we generate a set of queries and store the results of those queries. Note that the size of the dataset does not influence the performance of our approach but only the size of the Q set. Thus, we evaluate the parameters that influence this size.



(a) Varying k (b) Varying t (c) Varying $|Q|$

Figure 2: Diversity of the selected objects

Parameter	Values
k	$\mathbf{5}, 10, 15, 20, 25, 30$
$\|t\|$	$2, \mathbf{3}, 4, 5$
$\|Q\|$	$10, 25, \mathbf{50}, 75, 100$

Table 3
Diverse m Selection Problem $DmSQ$ our approach parameters.

Dataset: The dataset contains real data, which were obtained from factual.com and describes restaurants for 13 US states ($\approx$ 79K objects). In more details we collected restaurant that are annotated with their location. Moreover, for the collected restaurants we added textual description of the served food, mentioned as "cuisine". The number of distinct values of keywords for the cuisine is around 130 and each restaurant description may contain one or more keywords.
Evaluated Approaches: We compare the following approaches:

1. Random ($RAND$), selecting m points of interest randomly
2. Naive m Selection Query ($NmSQ$), a naive approach that takes into account the ranking position of the points of interest.
3. Diversified m Selection Problem ($DmSQ$), our approach to select m diverse points of interest.

Query generation: For each experiments we generate $|Q|$ Keyword kNN Query (KNQ) queries by using the following approach: for each query we randomly (uniformly) pick a latitude and longitude that falls in the area that each defined by the minimum and maximum values of coordinates of the points of interest in our dataset. The k is set to a given value per experiment. In order to make sure that at least one point of interest exists with the given keywords, the keywords are generated by picking a random (uniformly) point of interest and selecting at most $|t|$ keywords of the selected point of interest. All queries in each experiments have the same k and $|t|$ parameters, while the location and the keywords vary per query.
Experimental parameters: We vary the parameters of the Keyword kNN Query (KNQ) parameters in our experiments. The parameters are k the number of retrieved data objects and $|t|$ the number of given terms per query. In addition, for the Diverse m Selection Problem $DmSQ$ we vary as parameters the number of queries $|Q|$ that are stored in the log file, while the m parameter is set to 3. Table 3 overviews the parameters, while the default values are depicted with bold. Finally, the Haversine distance is used as a spatial distance in the experimental evaluation.

5.2. Experimental Result

In Figure 3, we illustrate the time required to identify the $m = 3$ points of interest using our $DmSQ$ algorithm. As expected, the number of retrieved data k increases the time for computing the $DmSQ$, since the number of the candidate objects increase.

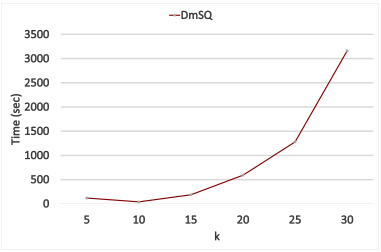


Figure 3: Time varying k

In the next set of experiments, we compared the three approaches based on the diversity ($div()$) of the m selected objects. We depict the diversity of the retrieved result set for the three different approaches, i.e., the value 1 is the maximum value and indicates high diversity.

In the first experiment we vary the parameter k ($|Q| = 50$, $|t| = 3$) and measure the diversity that should be as high as possible since we aim to diverse result sets. Figure 2a depicts the results of this experiment. We notice that $RAND$ and $NmSQ$ have similar results as none of those takes into account the similarity during the selection process. More interestingly, it seams that $NmSQ$ retrieves even less diverse points than $RAND$. This is because $NmSQ$ favours objects that appears in the result set of popular or similar queries. These objects even though they are popular, are interested only for a fraction of users. On the other hand, $DmSQ$ succeeded much larger diversity values, and as expected is not influenced by the k value.

In the second experiment we vary $|t|$ ($k = 5$, $|Q| = 50$) and the results are depicted in Figure 2b. We notice that the diversity decreases with t. The main reason is that the k objects retrieved by the Keyword kNN Query (KNQ) have a smaller spatial distance, since more objects satisfy the keyword criteria. Thus, the candidate objects are closer in spatial space leading to smaller values of diversity, i.e. higher similarity.

Finally, Figure 2c shows the results for varying $|Q|$ ($k = 5$, $|t| = 3$). Again, $DmSQ$ outperforms the other approaches and is not influenced by the value of Q, while the diversity of $RAND$ and $NmSQ$ is smaller in all cases.

To summarise, the experimental evaluation shows that in

all cases our approach manages to retrieve a set of m points of interest with high diversity.

6. Conclusions

In this paper, we address the challenge of identifying the most popular objects according to query log file, while also ensuring that select objects demonstrates a significant level of diversity. To this end, we propose a novel approach (Diversified m Selection Problem $DmSQ$) that retrieves a set of m points of interest that are popular and diverse at the same time. In our experimental evaluation, we study the performance of our approach using varying number of queries (Q), number of retrieved objects of interest (k) and number of querying words (t). We compare the diversity of the selected points against a set of randomly selected points and a a naive approach which takes into account only the popularity of the objects. Our experimental evaluation shows that in all cases the Diversified m Selection Problem $DmSQ$ succeeds to retrieve objects of high diversity.

References

[1] M. L. Yiu, X. Dai, N. Mamoulis, M. Vaitis, Top-k spatial preference queries, in: 2007 IEEE 23rd International Conference on Data Engineering, 2007, pp. 1076–1085.

[2] Z. Chen, L. Chen, G. Cong, C. S. Jensen, Location- and keyword-based querying of geo-textual data: a survey, The VLDB Journal 30 (2021).

[3] L. Chen, G. Cong, C. S. Jensen, D. Wu, Spatial keyword query processing: An experimental evaluation, Proceedings of the VLDB Endowment 6 (2013) 217–228.

[4] L. Chen, S. Shang, C. Yang, J. Li, Spatial keyword search: a survey, GeoInformatica 24 (2020).

[5] I. De Felipe, V. Hristidis, N. Rishe, Keyword search on spatial databases, in: 2008 IEEE 24th International conference on data engineering, IEEE, 2008, p. 656–665.

[6] X. Cao, G. Cong, C. S. Jensen, B. C. Ooi, Collective spatial keyword querying, in: Proceedings of the 2011 ACM SIGMOD International Conference on Management of data, 2011, pp. 373–384.

[7] Y. Tao, C. Sheng, Fast nearest neighbor search with keywords, IEEE transactions on knowledge and data engineering 26 (2013) 878–888.

[8] J. B. Rocha-Junior, O. Gkorgkas, S. Jonassen, K. Nørvåg, Efficient processing of top-k spatial keyword queries, in: Advances in Spatial and Temporal Databases, Lecture Notes in Computer Science, Berlin, Heidelberg, 2011, p. 205–222.

[9] H.-Y. Kwon, K.-Y. Whang, Scalable and efficient processing of top-k multiple-type integrated queries, World Wide Web 19 (2016).

[10] M. Drosou, E. Pitoura, Search result diversification, ACM SIGMOD Record 39 (2010) 41–47.

[11] M. R. Vieira, H. L. Razente, M. C. N. Barioni, M. Hadjieleftheriou, D. Srivastava, C. Traina, V. J. Tsotras, On query result diversification, in: 2011 IEEE 27th International Conference on Data Engineering, 2011, pp. 1163–1174.

[12] L. Qin, J. X. Yu, L. Chang, Diversifying top-k results, Proceedings of the VLDB Endowment 5 (2012) 1124–1135. doi:10.14778/2350229.2350233.

[13] Z. Yang, A. W.-C. Fu, R. Liu, Diversified top-k subgraph querying in a large graph, in: Proceedings of the 2016 International Conference on Management of Data, 2016, p. 1167–1182.

[14] O. Gkorgkas, A. Vlachou, C. Doulkeridis, K. Nørvåg, Finding the most diverse products using preference queries., in: 18th International Conference on Extending Database Technology, 2015, pp. 205–216.

[15] X. Ge, P. Chrysanthis, A. Labrinidis, Preferential diversity, in: Proceedings of the Second International Workshop on Exploratory Search in Databases and the Web, 2015, pp. 9–14.

[16] G. Kalamatianos, G. J. Fakas, N. Mamoulis, Proportionality in spatial keyword search, in: Proceedings of the 2021 International Conference on Management of Data, 2021, pp. 885–897.

[17] S. Maropaki, S. Chester, C. Doulkeridis, K. Nørvåg, Diversifying top-k point-of-interest queries via collective social reach, in: Proceedings of International Conference on Information and Knowledge Management, ACM, 2020, pp. 2149–2152.

[18] L. da Fontoura Costa, Further generalizations of the jaccard index, ArXiv (2021).

[19] G. Tsatsanifos, A. Vlachou, On processing top-k spatio-textual preference queries., in: 18th International Conference on Extending Database Technology, 2015, p. 433–444.