Using Summaries to Search and Visualize Distributed Resources Addressing Spatial and Multimedia Features

Daniel Blank · Andreas Henrich · Stefan Kufer

Abstract Summarization is an important means to cope with the challenges of big data. Summaries can help to achieve a first overview, they can be used to characterize subsets, they allow for the targeted access to data, and they build the basis for visualization techniques. In the present article, we point out the role of summaries as well as potential application scenarios. As examples, summarization techniques for spatial data (as an example for specific low dimensional techniques) and for general metric spaces (as a generic example with a broad spectrum of applications) are described. Furthermore, their use for resource selection and resource visualization in large distributed scenarios is outlined.

Keywords Resource description and selection · Distributed metric and spatial access methods · Big data summarization

1 Introduction

From an *information retrieval* perspective, dealing with "*big data*" often involves summarization techniques. One reason is that distributed retrieval solutions are inevitable at a certain scale. Thus, this article outlines resource description and selection techniques and their use for big data applications. These techniques can for example be applied in infrastructures with physically distributed nodes such as peer-to-peer (P2P) networks, cloud infrastructures, grid infrastructures, or sensor networks, to name only a few.

In particular, in this article, similarity search is considered where similar objects with respect to a given query object are to be retrieved. Many similarity search problems are modeled in metric spaces where the distance measure is a metric. Thus, metric space indexing techniques which do not rely on any further assumptions about the object representations allow for a variety of application fields. Examples are similarity search on business process models [23], malware detection [17], 3D object retrieval [7], text and multimedia retrieval, data compression, pattern recognition, machine learning, biomedical databases, statistical data analysis, and data mining [9, *ch.* 2], [29, *p.* 3f.]. In addition to general metric access methods (MAMs), specialized access methods for specific data types exist. As an example we will address spatial access methods (SAMs) and corresponding summarization techniques (see [26]) here.

In general, different criteria can be employed for the retrieval of media items such as text, timestamps, geographic footprints, and (low-level) audio or visual content information. Resource description and selection techniques for text data are for example proposed in [11]. They are not addressed here, nor do we focus on techniques for time and date information. The focus of this article is the outline of resource description and selection techniques for geospatial data as well as general metric data[1].

D. Blank (✉) · A. Henrich · S. Kufer
Media Informatics Group, University of Bamberg,
Bamberg, Germany
e-mail: daniel.blank@uni-bamberg.de

A. Henrich
e-mail: andreas.henrich@uni-bamberg.de

S. Kufer
e-mail: stefan.kufer@uni-bamberg.de

[1] Resource selection based on a single criterion, for example image content, is only a first step on the way to an effective retrieval system. When querying for multiple criteria, for example for an image with a particular content which was taken in a certain geographic region, criterion-specific resource rankings can be combined by applying a merging algorithm for ranked lists (*cf. e.g.* [2], [19]). Moreover, resource description and selection schemes can be designed which sup-

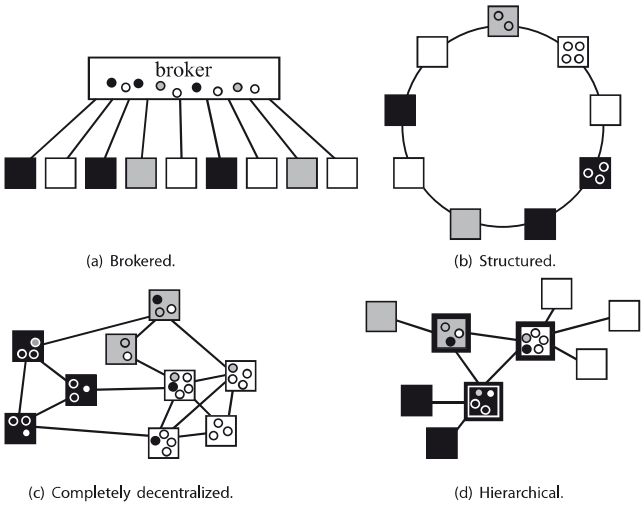

Fig. 1 Different P2P network architectures according to [24]; visualization inspired by [24, Fig. 2.1 on *p*. 7]. Squares □ denote peers; those with large black borders ◘ are super-peers. Circles ◯ visualize resource descriptions. Equal fill colors indicate a similar "thematic" focus of peers and/or resource descriptions

The remainder of the article is organized as follows. Different types of network architectures in which summaries can be applied are sketched in section 2. Techniques for the design of spatial as well as general metric resource descriptions are outlined in sections 3 and 4. Section 5 presents an evaluation of the introduced resource descriptions for the resource selection task using the example of geospatial data derived from Twitter. The applicability of resource descriptions for resource visualization as another important aspect in the context of big data applications is addressed in section 6, where a distributed image browser with images from Flickr is presented as an example. Finally, section 7 concludes this article.

2 Summary-based Network Architectures

Four types of general P2P architectures are identified in [24]. We adopt the classification from [24, *ch*. 2.1.1] to show the wide applicability of resource description and selection techniques in different network architectures.

Brokered architectures. Not only P2P IR systems, but also traditional distributed IR systems for federated text search rely on broker-based architectures. The descriptions of participating resources are administered by an information broker (see Fig. 1a). When a resource issues a query, this query is first sent to the broker. Based on the resource descriptions, the broker then decides which of the resources to contact. Usually, the broker sends addresses of promising resources to the enquiring resource which then issues the query to the selected resources and waits for their results.

Structured architectures. Network dynamics with resources frequently entering the system and/or updating their content can lead to network load a distributed IR system relying on global document indexing can hardly cope with. Instead of distributing a global document index, several authors thus propose the indexing of resource descriptions in a distributed index structure such as a distributed hashtable (DHT). Fig. 1b shows an example scenario where similar summaries are administered by the same peer applying some type of similarity preserving hash function.

Completely decentralized architectures. Completely decentralized P2P IR systems are *pure* P2P IR systems. They lack the presence of a central authority or super-peers. Semantic Overlay Networks (SONs) [10] fall into this group of systems. Every peer maintains two types of links to other peers. Peers with similar content are connected by short links and form communities (see Fig. 1c). Descriptions of peer content can be used in combination with a similarity measure to derive a peer's place within the network topology. To do so, clustering, classification, and gossiping techniques are applied [12]. Long links are established between different communities to ensure the connectedness of the resources. During query execution, the query is forwarded to the most promising communities by multi-hop query routing.

A subtype of completely decentralized P2P IR systems assumes that every peer knows the resource descriptions of all other peers in the same (sub)net (see for example the four white peers in Fig. 1c which form such a subnet). Approaches belonging to this group are for example PlanetP [11] and its extension Rumorama [25]. PlanetP assumes that a peer knows the resource descriptions of all other peers in the system. Rumorama tries to assure scalability by building hierarchies of PlanetP networks. In Rumorama, every peer sees a portion of the network as a single, small PlanetP network and furthermore maintains connections to other peers that see other small PlanetP networks.

Hierarchical architectures. Hierarchical P2P IR architectures (see [28] and the visualization in Fig. 1d) are designed to overcome some limitations of other types of P2P IR systems. Super-peers, as opposed to (normal) peers, make use of increased capabilities such as storage capacity, processing power, network bandwidth, or availability. Often, concepts known from other types of P2P IR systems are extended and adapted by hierarchical P2P IR architectures such as for example resource description and selection techniques.

Techniques from the first three types of network architectures outlined in this section can be applied and combined in hierarchical architectures. There are for example hierarchies

port content-based search and in addition preserve the geographic distribution of the images by integrating both content-based as well as geographic search criteria (*cf. e.g.* [14] which combines text and geographic information). However, such aspects are out of the scope here.

of brokers, super-peer networks based on structured P2P IR architectures, and semantic overlay networks where peers with similar content are assigned to the same super-peer (for references see [3]).

In the above-mentioned scenarios, resource descriptions can be identified at different levels [24, *p.* 39–42]: resource descriptions of peers, resource descriptions of super-peers, and resource descriptions of (super-)peer neighborhoods. Furthermore, these scenarios can be seen as prototypical for the broad spectrum of distributed data management settings. They clarify the role of summaries describing the content maintained on different resources in these settings.

The following sections will now outline exemplary resource description and selection techniques applicable in the stated network architectures. Sect. 3 starts with spatial summaries as an example for specific low-dimensional features before metric summaries are addressed in Sect. 4 as an example for generic summaries applicable in a broad spectrum of situations.

3 Spatial Summaries

Hereafter, geo-coordinates are treated as plate-carrée-projected data points, meaning the lat/long-coordinates correspond to x/y-coordinates in a 2D Cartesian coordinate system. The Euclidean distance is used for distance calculations, since the usage of distance measures better suited for distance calculations on the earth's surface does not show noticeable changes as evaluated in [4]. Thus, the resource description problem is reduced to encoding a set of two-dimensional spatial data points effectively (accurate description) and efficiently (compact storage), which is a more general, extensive scenario. Previous work ([20–22]) has investigated several techniques, which shall be briefly recapitulated:

Geometric Approaches. One or more geometric shapes enclosing all of the resource's data points are calculated as a representation. One simple, well-known example would be the *Minimum Bounding Rectangle* (MBR), enclosing all data points in a minimum sized rectangle.

Space Partitioning Approaches. Space partitioning approaches globally segment the data space into a certain number of identifiable subspaces or cells. Global segmentation means it is the same for all resources. The summaries' expressiveness is gained from storing information about how the single cells are populated with data points. It is most beneficial to encode binary information (whether the resource contains at least one data point in a specific cell), but frequency information is also an option (see [20]). A simple example would be Grid_r, a regular grid segmenting the data space with r grid rows and $2 \cdot r$ grid columns.

Hybrid Approaches. Hybrid approaches combine two different approaches: one technique is used for building the foundation of the description; the second one is used for refining this foundation. Often, it is beneficial to combine a geometric approach and a space partitioning approach. Both categories are employable as the foundation as well as the refinement. An example for a hybrid approach would be K-D-MBR_n^b described later.

In the following, we briefly present the different spatial summaries which are evaluated in section 5.

The MBR is a well-known standard technique, which will serve as a baseline approach in our evaluation. For each resource, the minimal rectangle containing all data points is calculated. In order to encode a rectangle, four values specifying the lower left corner and the upper right corner have to be stored. Each value is encoded in float precision (32 bit).

The $\text{RecMAR}_{k,s}$ approach is based on an algorithm from [1] which has been adjusted to point data. Initially, a single MBR is disassembled into two so-called Minimum Area Rectangles (MARs), which are the two rectangles covering minimal area whilst containing all data points. This process is repeated recursively by choosing the biggest MAR in the current set of MARs and disassembling it into two, again, until an amount of k MARs has been computed or a certain stopping criterion s is reached (see [20] for details). For the summaries, the MARs are consecutively encoded in float precision.

K-D-MBR_n^b is a hybrid technique which takes a k-d tree like space partitioning as a foundation and refines occupied cells with quantized MBRs to further confine the locations of a resource's data points. With the k-d approach, the data space is segmented into rectangular cells of adaptable sizes (in contrast to the equally sized cells of Grid_r). The specific space partitioning is learned from training data, whose distribution has to resemble the collection's distribution for optimal results. At the beginning of the training phase, the data space consists of a single cell or bucket. Next, training data points are continuously inserted into the bucket until a bucket overflow occurs, after which the bucket is split. Afterwards, the training data point insertion is pursued. The whole process continues until an amount of n buckets has been reached. Different strategies for determining split dimensions and split positions are applicable; we resort to cyclically altering the split dimension and the cell's middle for the split position. As refinement, one quantized MBR is calculated for each cell containing data points. The accuracy of the quantization is controlled by the parameter b, determining the number of bits used to encode one of the four necessary MBR values. By applying quantization, we lose some accuracy (the quantized rectangles are slightly bigger than the 'real' MBRs, depending on the parameterization) but safe a lot of storage space compared to using 32 bit float values. Bit vectors are used as a summary, primarily encoding binary information about cell occupancy for the foundation; after a '1' for an occupied cell, $4 \cdot b$ bits follow to encode the cell's associated quantized MBR.

Ranking The MBR, RecMAR$_{k,s}$ and K-D-MBR^b_n summaries all describe one or several rectangular areas which contain data points. Hence, their ranking algorithm is the same. First, the rectangles representing a resource are reconstructed. Then, the minimum distance between each rectangle and the query location is calculated. Afterwards, for each rectangle, the distance information and the area covered by the rectangle are stored in a so-called R-Entry. For the ranking process, the R-Entries are sorted by (1) distance in ascending order and (2) —in case of equal distances— by area covered in ascending order (assuming a smaller area indicates a higher point density and therefore is more likely to include relevant[2] data points than a larger area at the same distance). To determine the ranking of two resources, the sorted R-Entries are compared one after another using distance as 1st criterion and area covered as 2nd criterion (same way as for the resource-individual R-Entry sorting).

4 Summaries for General Metric Spaces

In contrast to summaries designed explicitly for spatial data, in general metric spaces a direct representation of subsets of the data space based on a coordinate system—for example using MBRs—is not applicable. In fact, we can only rely on the given metric and calculate distances between objects and between objects and reference objects.

MAMs can be reviewed when looking for summaries for general metric spaces. These methods apply summaries to prune subspaces during query processing. Hetland [16, *Sect.* 9.4] distinguishes two kinds of MAMs: MAMs based only on *pivoting* store distances from data objects to reference objects (so called pivots) and prune data objects during search through pivot filtering based on the precomputed object-to-pivot distances. MAMs using *aggregation* [16, *p.* 203f.], occasionally in addition to pivoting, structure the feature space into multiple regions in order to prune non-relevant regions during search. The same concepts—pivoting and aggregation—can be used to distinguish resource descriptions.

Pivoting might be a good idea in a centralized scenario where distance calculations (on high dimensional data) are the most expensive steps and the maintenance of object-to-pivot distance matrices is affordable. In a distributed setting, however, aggregation techniques appear more promising. Hence, in the following, we describe two types of resource descriptions which are based on aggregation: UFS and DFS.[3]

For UFS$_{n,cc}$ (Ultra Fine-grained Summaries), the data space is segmented globally by n preselected reference objects, which invoke a Voronoi-like space partitioning onto the data space. Each of a resource's data objects is located in the cell of the closest reference object according to the given metric. The selectivity for this approach comes from utilizing a large number of reference objects. Likewise K-D-MBR^b_n, the distribution of the reference objects has to resemble the distribution of the data collection's data objects. The parameter cc determines how many reference objects are considered during ranking. As a summary, binary information about cell occupancy is stored in a bit vector of length n. UFS$_{n,cc}$ has been designed for the application in general metric spaces and is well suited for approximate search. However, it is inappropriate for precise search in high-dimensional spaces [3].

DFS$^b_{n,cc}$ (Distance enhanced UFS) also takes a Voronoi-like space partitioning as a foundation. It is enhanced with quantized distance information for a hypersphere containing all of the cell's data objects for each occupied Voronoi cell. Parameter b controls the number of bits utilized for encoding one distance value. The hypersphere radius for a cell is determined by the maximum distance between the reference object of a cell and any of its associated data objects. Since this information is quantized, it is an approximated hypersphere which is bigger than the 'real' hypersphere. Details of the quantization are outlined in [22].

Ranking Like the rectangle-based techniques, the Voronoi-like techniques apply the same ranking algorithm. The n reference objects c_i are sorted in ascending order to the query object. The first element of the sorted list L corresponds to the reference object being closest to the query. It is the center of the so-called query cluster. If resource r_a administers documents in this query cluster while r_b does not, r_a is ranked higher than r_b. If both resources feature the same value for the query cluster, the next element out of L is chosen and both resources are ranked according to their summary values for this very cluster. This procedure continues until a decision favoring one of the resources can be made or the first cc elements of L are considered, resulting in a random decision. For DFS$^b_{n,cc}$, the distance information encoded in the summaries is used for pruning purposes only.

[2]Note that we may use the term (non-)relevant differently from traditional IR, where it is strongly related with the information need concept. We speak of (non-)relevant database objects to indicate that they are (not) part of the final query result. Similarly, (non-)relevant feature space regions or resources do (not) contain database objects from the final result. This interpretation corresponds with for example [27].

[3]An overview of different summarization techniques both using pivoting and/or aggregation is given in [3].

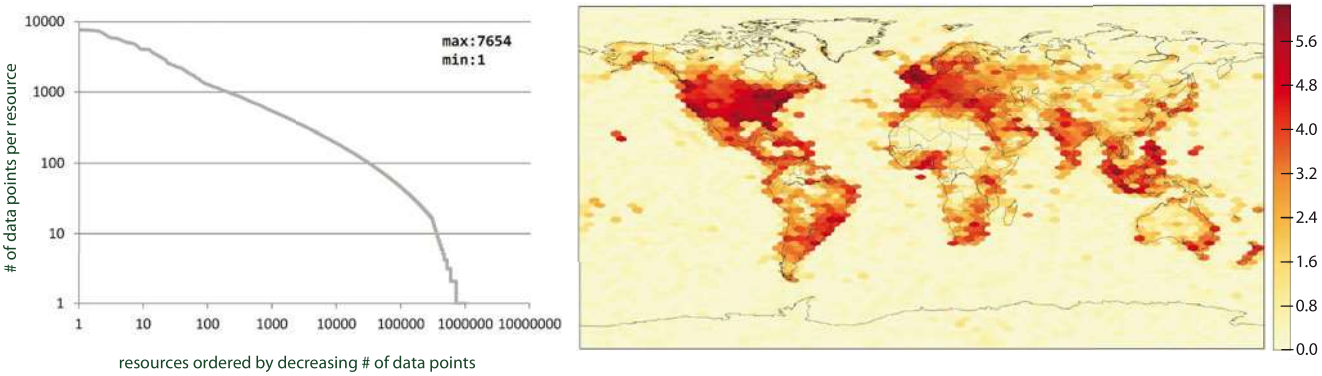

Fig. 2 On the left, the allocation of data points to resources for the collection (note that the axes are logarithmic scaled). On the right, the distribution of the data points in the data space for the collection. The values on the respective legend are $log_{10}(x+1)$ scaled, so the number of data points per bin is calcluated by $x = 10^n - 1$. For example, $n = 4.0$ results in $x = 9999$

5 Resource Selection

In this section, we present a comparison of the resource selection effectiveness and the efficiency of the resource description approaches outlined in the two previous sections. As a common test scenario, we apply all summaries to geospatial data, although UFS and DFS are not limited on this scenario. For an extensive evaluation of summaries for general metric spaces, see [3, 5].

5.1 Data Collection and Experimental Setting

From August, 25th 2014, to September, 1st 2014, we crawled geo-referenced tweets from Twitter, picked of in time windows of 3 min each. The tweets have been restricted to the English language (determined by the Twitter API). In total, we gathered 26,767,783 tweets from 2,620,571 users. The first ten percent of time slices are used as a source for training data and query points, the rest is utilized as test data. After this separation, 23,539,714 tweets from 2,491,785 users remain for testing. Since we focus on the geospatial aspects, only the geo-references of the tweets are considered in our experiments, the textual content is left out. Hence, the summaries delimit areas which contain geo-references/data points (i.e. the lat/long-coordinates assigned to the tweets). In order to assign data points to resources, we assume each user operates a resource of his own, hence there are 2,491,785 resources in our experiments. Figure 2 depicts the distribution of data points to resources, which is very skewed: Only few resources maintain many data points whilst the majority of resources holds only few data points[4]. In Fig. 2, the geographic distribution of the data points is also shown. Due to the focus on the English language, the countries' development, and the population numbers, it is no surprise that most data points are located in the USA and on the British Isles. Nevertheless, some other regions—such as South East Asia, India, and South America—also have a fair share of the total volume.

[4]In order to fit into a spreadsheet, the allocation has been sampled. The first 300,000 resources show the orginal values. Afterwards, three resources are sampled into one value, which is a very accurate depiction of the real distribution, since the 300,001st biggest resource only has 16 data points, degrading one-by-one to one for resource 2,491,785.

MBR		(none)
$RecMAR_{k,s}$	k	3|6|9
	s	1|0.01|0.0001|0.00005
$KD\text{-}MBR^{b}_{n}$	n	512|2048|8192
	b	3|4|6
$UFS_{n,cc}$	n	512|2048|8192
	cc	16|64|256|all
$DFS^{b}_{n,cc}$	n	512|2048|8192
	cc	16|64|256|all
	b	3|4|6

k = # of MARs
s = **s**topping criterion (max. dist of any data point to its rectangle center)
n = number of subspaces
b = # of **b**its for quantization
cc = # of **c**entroids **c**onsidered

Fig. 3 Parameter variation for the different techniques

For the query points, we randomly chose 50 data points out of the training data (for example, we know of a tweet posted at the *Reichstag* in Berlin and want to retrieve the 50 tweets out of the network which have been posted closest to the *Reichstag*). Both the reference points for the Voronoi-based approaches as well as the training data for K-D-MBR^{b}_{n} are randomly chosen from the training data, too.

The variation of parameters for the different techniques is shown in Fig. 3. All possible combinations are tested. We evaluate both *selectivity* (avg. fraction of resources contacted

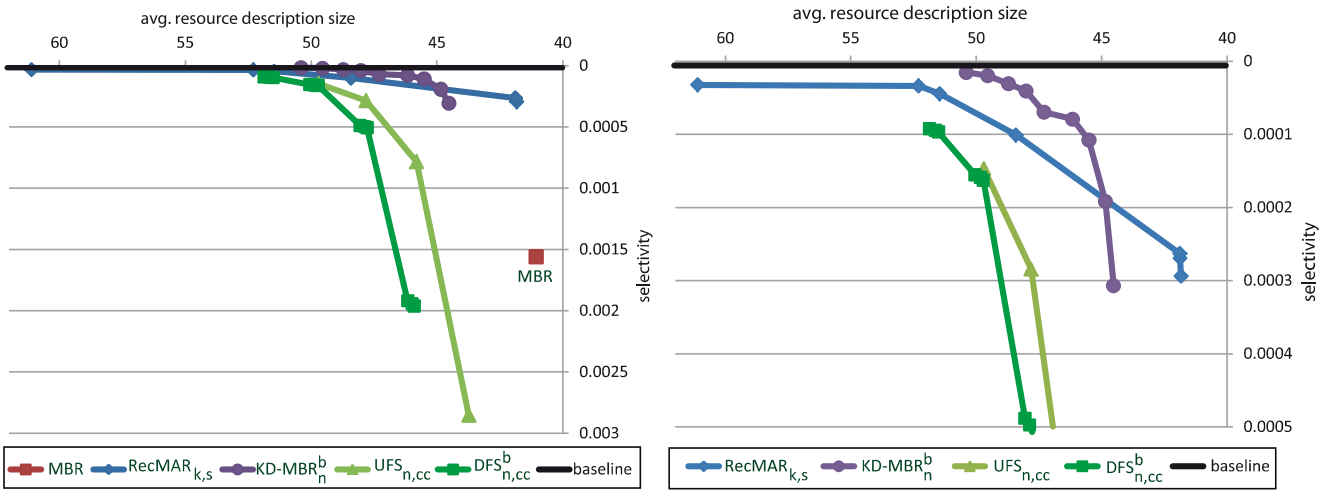

Fig. 4 Result Skylines for the different approaches. On the right a more detailled view of the best results

in order to retrieve the top 50 data points for each query) and *avg. resource description sizes*. Generally, there are two options for transferring data: Either represent the resource by the summary computed for it; or directly transfer the data points of the resource in case they require less storage space than the summary. Additionally, the data sizes might be reduced by compression. We use Java's `gzip` compression with default parameters to reduce the amount of data in case it is beneficial. Thus, there are four options for a resource to be represented:

- 1st and 2nd: Represented by a summary (zipped: *sumz*; non-zipped: *sumnz*).
- 3rd and 4th: Directly represented by its data points (zipped: *dtz*; non-zipped: *dtnz*).

In any case, we require 27 byte serialization overhead and 1 byte for encoding which data transfer option has been chosen in addition to the actual resource data. The ranking process is affected neither for the spatial nor for the metric resource descriptions by the possibility of a direct representation. For the spatial descriptions, each data point is treated as a rectangle with an area of 0. For the metric descriptions, the usual cell occupancies (plus the non-quantized maximum distance for each cell in case of $\text{DFS}^b_{n,cc}$) are utilized.

For choosing the best parameterizations to represent a technique, we apply the Skyline operator [6]. The measurements for the different parameterizations of a certain technique are represented as two-dimensional data points with the dimensions selectivity (*sel*) and avg. resource descriptions size (*rds*). Smaller values are preferable for both dimensions. Generally, the Skyline of a set of points is formed by the points which are not dominated by other points. A point dominates another point if it is as good or better in all dimensions and better in at least one dimension [6]. Thus, for a certain technique, a parameterization x dominates a parameterization y if ($x.sel \leq y.sel$ and $x.rds < y.rds$) or ($x.sel < y.sel$ and $x.rds \leq y.rds$). For inter-technique comparisons, we collate the Skylines of the different techniques.

We conduct precise k-nearest-neighbor ($k$nn) queries ($k = 50$) in our experimental runs. As algorithm, a range query with decreasing query radius is applied. The query range is set to the distance of the current 50th neighbor, hence infinite at start. After an initial ranking based on the resource descriptions and the respective ranking algorithms, the $k$nn algorithm queries the resources round wise (in chunks of ten resources) for their most relevant data points, updating the top-k result and the query radius if necessary (which is gradually reduced as more and more resources are queried for their NNs). Due to their descriptions definitely irrelevant resources are pruned in the meanwhile. The algorithm ends when all resources have been queried or pruned. See [21] for more details concerning the $k$nn-algorithm.

5.2 Experimental Results

On average, $12.91\overline{9}$ resources maintain relevant data points, resulting in an optimum selectivity of $12.91\overline{9}/2,491,78 = 5.185E{-}6$. This baseline is indicated as the top black line in the Skyline diagrams (see Fig. 4).

For the simple MBR, it shows that the avg. resource description sizes are the smallest, but selectivity is far worse compared to other techniques. Especially $\text{RecMAR}_{k,s}$ only requires few additional storage while offering very big gains in selectivity (see Fig. 4).

In general, MBR is not too bad in terms of selectivity when compared to $\text{UFS}_{n,cc}$ or $\text{DFS}^b_{n,cc}$ utilizing a low amount of reference points. This is due to the distribution of the data

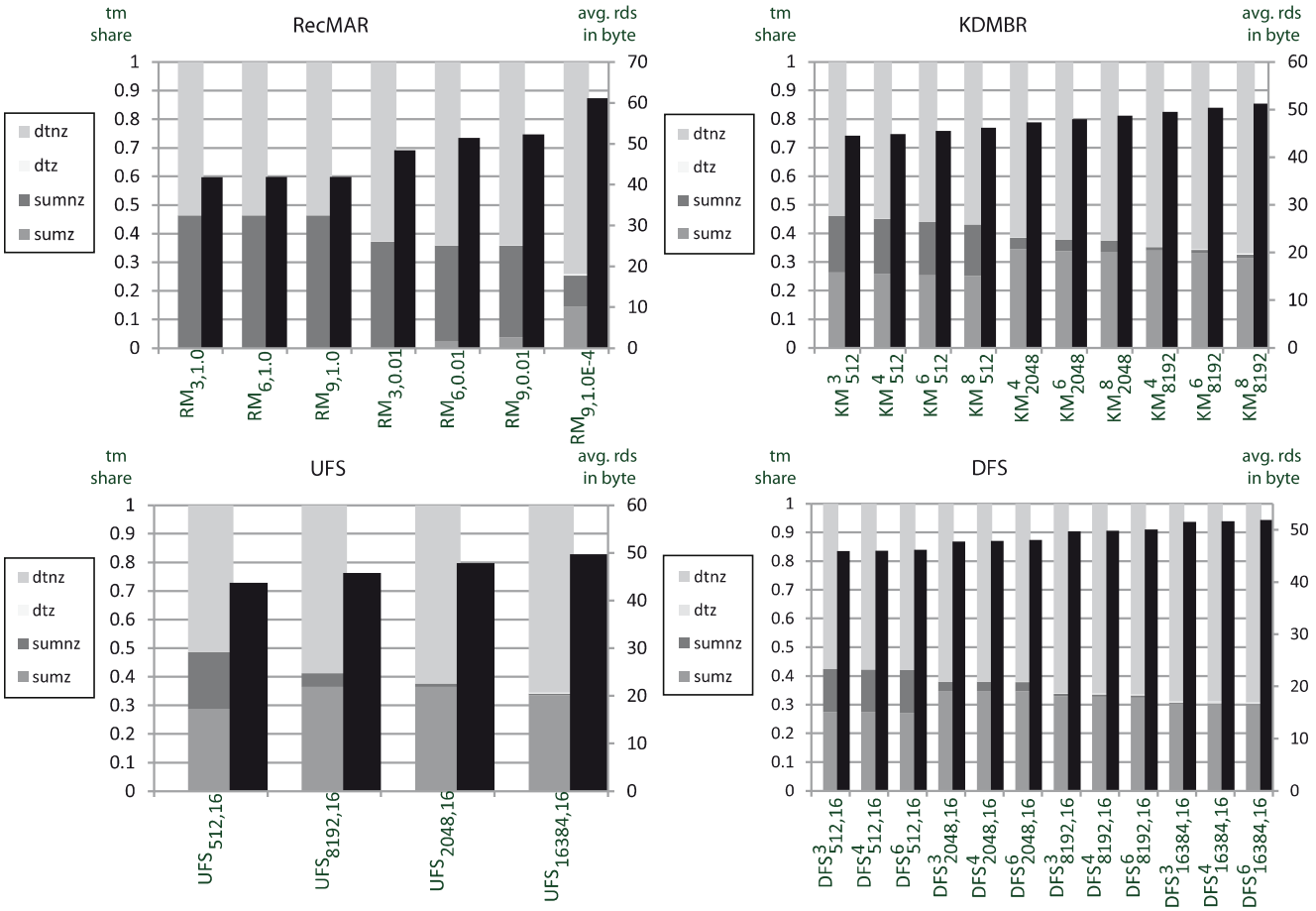

Fig. 5 Percentage share of the optimum transfer method (*tm share*) for the different techniques and their respective parameterizations (*grayscale columns* on the primary axis); the secondary axis shows the respective average resource description sizes (*black columns*). For MBR, the *tm share* is 47.67% *sumnz* and 52.33% *dtnz* at an *avg. rds* of 41.06 byte

points resulting from the way of creating the data collection. For one thing, twittering obviously takes place in the daily routine and therefore mostly from home/work. For another thing, since the data has been collected for only 1 week, it is apparent that most people do not get to travel very far around during 1 week (compared to the collection of geo-referenced Flickr images we used in previous work, for example [21], which includes data collected over several years and where the resource data hence is generally more spread, resulting in a worse MBR performance in relative terms). Hence, the point clouds of the resources usually only have a rather narrow geographic spread, facilitating the usage of rectangles for enclosing data point extents. For the Voronoi-based techniques, an amount of for example 512 reference points is not enough to effectively distinguish between relevant and irrelevant resources out of the population of 2,491,785 resources. Even though the reference point distribution has been adjusted to fit the collection, each Voronoi cell features data of far too many different resources.

The superiority of using rectangle-based techniques for the given data also becomes obvious in the Skylines, where $\text{RecMAR}_{k,s}$ and especially K-D-MBR^b_n clearly dominate $\text{UFS}_{n,cc}$ and $\text{DFS}^b_{n,cc}$, especially for low avg. resource description sizes (see Fig. 4). For bigger sizes, the gap diminishes a bit since compression comes favorably into play for $\text{UFS}_{n,cc}$ and $\text{DFS}^b_{n,cc}$ (see Fig. 5). $\text{RecMAR}_{k,s}$ starts as the most dominant technique for low avg. resource description sizes, but soon is overtaken by K-D-MBR^b_n since the latter offers higher peak selectivity and higher entropy summaries benefiting from compression (see Fig. 5, percentage share for the different data transfer options) due to its computation design as well as its summary design.

Comparing $\text{UFS}_{n,cc}$ and $\text{DFS}^b_{n,cc}$, the inclusion of distance information per occupied cell for optimized pruning does not pay off since $\text{UFS}_{n,cc}$ is dominant, at least for smaller sizes. For bigger sizes, a convergence can be observed. $\text{DFS}^b_{8192,16}$ has about the same selectivity and avg. resource description size as $\text{UFS}_{16384,16}$. Due to the added information, peak selec-

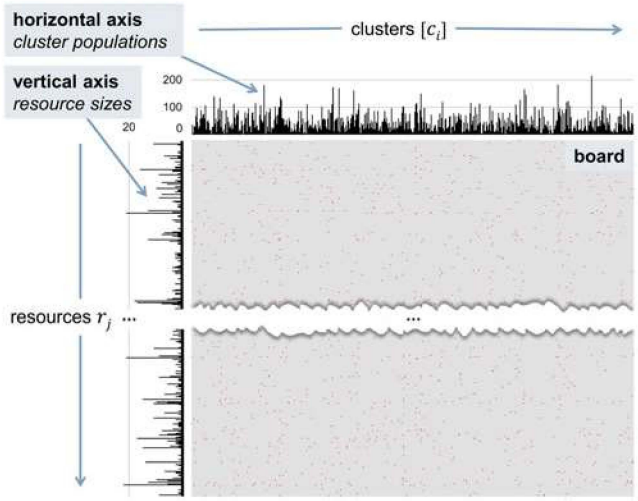

Fig. 6 Overview of our visual analytics interface

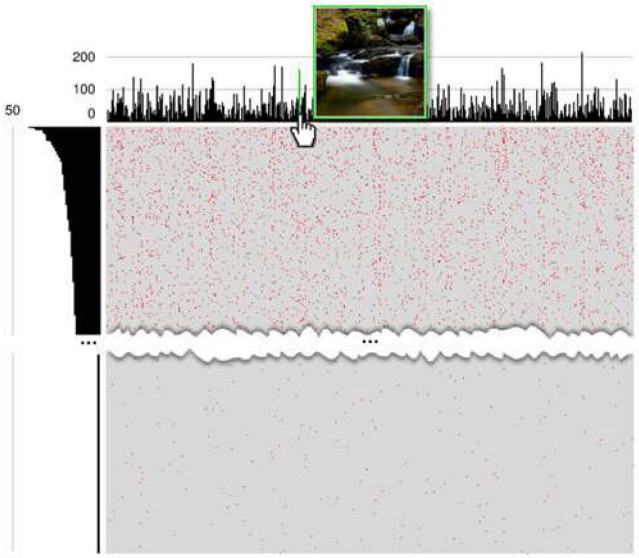

Fig. 7 Selecting a center as a query object after ranking the resources by estimated resource size

tivity of $DFS^{b}_{n,cc}$ is higher than for $UFS_{n,cc}$, but on the other hand still is significantly outperformed by the two sophisticated rectangle-based techniques, which are the preferable choices for the investigated data collection.

6 Resource Visualization

An important aspect of the resource descriptions and the corresponding resource selection step is that both can be visualized in a meaningful way. This opens doors for visual analytics.

To give a tangible example, we exemplarily outline the concept of a visual analytics interface proposed in [3, 15] which gives additional insights in the resource selection process and which can visually support the resource selection task. The visual interface is sketched in Fig. 6. It has a minimalistic design consisting of a board and a horizontal and a vertical axis. In the following, we assume UFS resource descriptions. However, the interface is by no means restricted to this particular summary type. In addition, the example is based on Flickr image data maintained on the resources and summarized based on low-level features[5], in contrast to the Twitter messages summarized based on geospatial data addressed in the previous section.

The resource descriptions of all participating resources are visualized on the board. To keep it clean without any additional overhead, a single row of pixels on the board is reserved for visualizing a particular resource description. The horizontal axis on top of the interface captures the distribution of the number of populated UFS bins. The histogram shows total cluster populations, that is per cluster, the number of summary bins containing at least one image representation aggregated over all resources. Clusters arise from the partitioning of the resource description technique that is used. In case of UFS, the Voronoi-like space partitioning is applied. On the other hand, the vertical axis on the left captures the distribution of the number of images of the resources (i.e. the resource sizes). Here, they are estimated based on the binary UFS summary bin values.

For UFS, a highlighted pixel on the board represents an occupied summary bin, that is, a summary bin value set to 1. When a summary bin is set to 0, the pixel representing the summary bin is inactive. For other summary types, graduated pixel colors could indicate cluster counts or similarity/distance values.

Figure 7 shows a ranking by resource size where the top part of the board contains by far more highlighted pixels than the bottom part. Also the estimated resource size distribution on the vertical axis indicates the ranking by the number of summary bins set to 1.

If the cluster centers are derived from real data objects, which is common in metric space indexing, the centers can be visualized for example by hovering over the horizontal axis (see Fig. 7). Cluster centers can be selected as query images for issuing similarity queries.

Figure 8 shows the visualization of the resource ranking step when using the ranking mechanism of UFS proposed in [13]. The pixel order on the board from left to right changes as clusters are rearranged. This can also be noticed when looking at the horizontal axis.

[5]We use the images of the MIRFLICKR-25000 collection [18]. CEDD [8] features are extracted and compared using the Hellinger distance. Images are assigned to resources by the Flickr user ID.

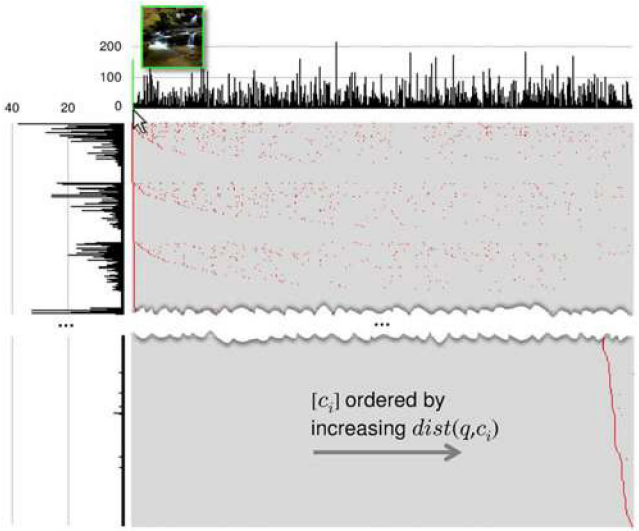

Fig. 8 The resource ranking in case of UFS

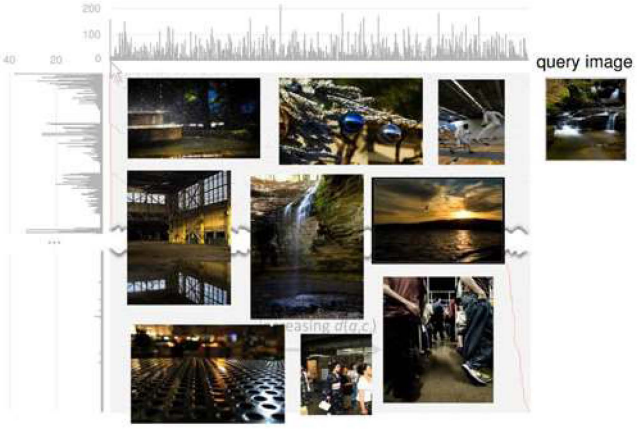

Fig. 9 An exemplary query result

The behaviour of the UFS ranking is shown on the board in Fig. 8 where it can be perceived that roughly spoken a bigger amount of populated clusters with centers close to the query lead to higher ranking positions. The cluster order is determined by the list L and thus the distance from a cluster center to the query object. An example query result is shown in Fig. 9.

Resource descriptions can be visualized in different ways. If images are for example tagged with geographic locations, it is possible to provide an overview of the geographic distribution of a resource's image collection. This can be achieved by visualizing geographic resource descriptions (see Fig. 10 where UFS summaries are also used for the geographic domain).

In addition, we would like to emphasize that the interface allows to focus on certain characteristics of the data collection. It is possible to assess special parts of the data collection such as for example resources with the least similar images which are found at the bottom part of the resource ranking as shown in Fig. 11.

Following the general design paradigm, clicking on a bar in the vertical axis, and thus selecting a certain resource, triggers a search for similar resources. To do so, resource descriptions themselves are perceived as feature objects and are compared with the query. This is in contrast to the selection of a certain cluster center from the horizontal axis as a query image. Thus, two different application modes can be supported by the sketched resource selection interface—searching for similar images to a query image and searching for similar resources/collections to a given image collection.

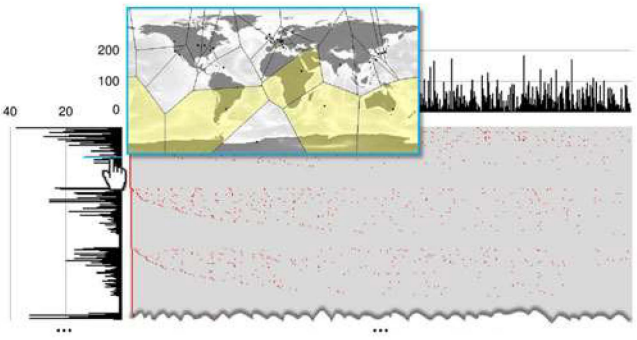

Fig. 10 Visualizing a second type of resource descriptions

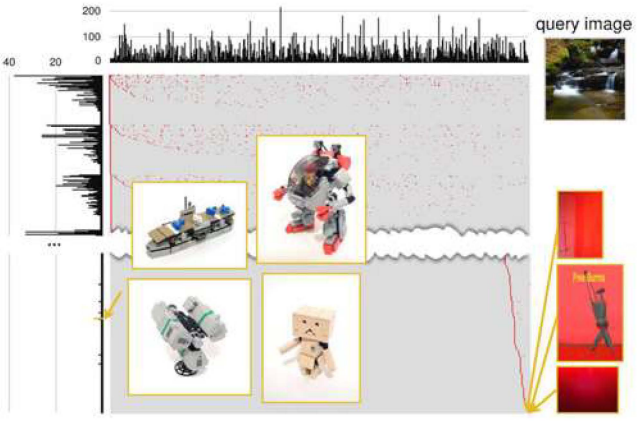

Fig. 11 Analyzing the bottom part of the resource ranking

7 Conclusion

In this article, we gave an overview of resource description techniques and their application scenarios. Effective and efficient resource descriptions are essential in many application areas. We can distinguish specialized techniques, for example optimized for spatial data and generic approaches. Besides applications in search and clustering scenarios, also visualization techniques allowing for the exploration of data sources can be based on such resource descriptions. Hence, powerful summarization techniques are an important tool to master the challenges of big data.

References

1. Becker B, Franciosa PG, Gschwind S, Ohler T, Thiemt G, Widmayer P (1991) An optimal algorithm for approximating a set of rectangles by two minimum area rectangles. Tech. rep., University of Freiburg, Freiburg
2. Belkin NJ, Kantor P, Fox EA, Shaw JA (1995) Combining the evidence of multiple query representations for information retrieval. Inf Process Manage 31:431–448
3. Blank D (2015) Resource description and selection for similarity search in metric spaces. University of Bamberg Press, Bamberg
4. Blank D, Henrich A (2012) Describing and selecting collections of georeferenced media items in peer-to-peer information retrieval systems. In: Díaz L, Granell C, Huerta J (eds) Discovery of Geospatial Resources: methodologies, technologies, and emergent applications. IGI global, Hershey
5. Blank D, Henrich A (2013) Resource description and selection for range query processing in general metric spaces. In: 15. Fachtagung Datenbanksysteme für Business, Technologie und Web, pp 93–112. GI, Magdeburg
6. Börzsönyi S, Kossmann D, Stocker K (2001) The skyline operator. In: Proceedings of the 17th International Conference on Data Engineering. IEEE Computer Society, Washington DC, pp 421–430
7. Bustos B, Keim D, Saupe D, Schreck T (2007) Content-based 3d object retrieval. IEEE Comput Graphics Appl 27:22–27
8. Chatzichristofis SA, Boutalis YS (2008) CEDD: Color and Edge Directivity Descriptor: a Compact Descriptor for Image Indexing and Retrieval. In: Proc. of the 6th Intl. Conf. on Computer Vision Systems. Springer LNCS 5008, Berlin, pp 312–322
9. Chávez E, Navarro G, Baeza-Yates R, Marroquín JL (2001) Searching in Metric Spaces. ACM Comput Surv 33:273–321
10. Crespo A, Garcia-Molina H (2005) Semantic overlay networks for p2p systems. In: Proc. of the 3rd Intl. Workshop on Agents and Peer-to-Peer Computing. Springer LNCS 3601, Berlin, pp 1–13
11. Cuenca-Acuna FM, Peery C, Martin RP, Nguyen TD (2003) PlanetP: using gossiping to build content addressable peer-to-peer information sharing communities. In: Proc. of the 12th Intl. Symp. on High Performance Distributed Computing. IEEE, Seattle, pp 236–246
12. Doulkeridis C, Vlachou A, Nørvåg K, Vazirgiannis M (2010) Distributed semantic overlay networks. In: Shen X, Yu H, Buford J, Akon M (eds) Handbook of Peer-to-Peer Networking, Part IV. Springer Science+Business Media, Berlin, pp 463–494
13. Eisenhardt M, Müller W, Henrich A, Blank D, El Allali S (2006) Clustering-based source selection for efficient image retrieval in peer-to-peer networks. In: Proc. of the 8th Intl. Symp. on Multimedia. IEEE, San Diego, pp 823–830
14. Hariharan R, Hore B, Mehrotra S (2008) Discovering GIS sources on the web using summaries. In: Proc. of the 8th Joint Conf. on Digital Libraries. ACM/IEEE, Pittsburgh, pp 94–103
15. Henrich A, Blank D (2012) Summarizing data collections by their spatial, temporal, textual and image footprint: techniques for source selection and beyond. Keynote Talk at the DFG SPP 1335 Text Workshop on Scalable Visual Analytics (held by: A. Henrich). Leipzig. November 15, 2012. https://www.uni-bamberg.de/fileadmin/uni/fakultaeten/wiai_lehrstuehle/medieninformatik/Dateien/Publikationen/2012/Henrich-Leipzig-2012.pdf (last visit: 8.10.2014)
16. Hetland ML (2009) The basic principles of metric indexing. In: Coello CAC, Dehuri S, Ghosh S (eds) Swarm intelligence for multi-objective problems in data mining, chap. 9. Springer, Berlin, pp 199–232
17. Hu X, Chiueh Tc, Shin KG (2009) Large-scale malware indexing using function-call graphs. In: Proc. of the 16th Intl. Conf. on Computer and Communications Security. ACM, New York, pp 611–620
18. Huiskes MJ, Lew MS (2008) The MIR Flickr Retrieval Evaluation. In: Proc. of the 1st Intl. Conf. on Multimedia Information Retrieval. ACM, New York, pp 39–43
19. Ilyas IF, Beskales G, Soliman MA (2008) A survey of top-k query processing techniques in relational database systems. ACM Comput Surveys 40:11:1–11:58
20. Kufer S, Blank D, Henrich A (2012) Techniken der ressourcenbeschreibung und -auswahl für das geographische information retrieval. In: Proc. of LWA Workshop. Dortmund. https://www.uni-bamberg.de/fileadmin/uni/fakultaeten/wiai_lehrstuehle/medieninformatik/Dateien/Publikationen/2012/kufer_2012_techniken.pdf (last visit: 1.10.2014)
21. Kufer S, Blank D, Henrich A (2013) Using hybrid techniques for resource description and selection in the context of distributed geographic information retrieval. In: Proc. of the 13th Intl. Symp. on Advances in Spatial and Temporal Databases. Springer LNCS 8098, Munich, pp 330–347
22. Kufer S, Henrich A (2014) Hybrid quantized resource descriptions for geospatial source selection. In: Proc. of the 4th Intl. Workshop on Location and the Web. ACM, Shanghai, 17–24
23. Kunze M, Weske M (2011) Metric trees for efficient similarity search in large process model repositories. Business Process Manag Workshops 66:535–546
24. Lu J (2007) Full-text federated search in peer-to-peer networks. Ph.D. thesis, Language Technologies Institute, School of Computer Science, Carnegie Mellon University. CMU-LTI-07-003
25. Müller W, Eisenhardt M, Henrich A (2005) Scalable summary based retrieval in P2P networks. In: Proc. of the 14th Intl. Conf. on Information and Knowledge Management. ACM, Bremen, pp 586–593
26. Samet H (2006) Foundations of Multidimensional and metric data structures. Morgan Kaufmann, San Francisco
27. Skopal T, Lokoč J, Bustos B (2012) D-cache: universal distance cache for metric access methods. IEEE Trans Knowl Data Eng 24:868–881
28. Yang B, Garcia-Molina H (2003) Designing a super-peer network. In: Proc. of the 19th Intl. Conf. on Data Engineering, pp 49–60. IEEE
29. Zezula P, Amato G, Dohnal V, Batko M (2006) Similarity Search: The Metric Space Approach. Springer New York, Inc., Secaucus